Algorithm selection

To begin, we constructed a simple model baseline to produce a reasonable snow water equivalent (SWE) prediction, as measured by root-mean-square error (RMSE). From the baseline we iterated to more complicated models that improved SWE prediction. The first baseline used the historical mean for every month and region pair in the train dataset as the SWE prediction for the relevant prediction dates and locations. The historical mean baseline achieved an RMSE of 8.1461 on the initial Test dataset. This demonstrated that SWE is heavily influenced by spatio-temporal properties. Even though this is intuitive it was helpful to get empirical feedback without resorting to complicated modeling efforts.

The importance of time and geographic features suggested a spatio-temporal variational gaussian process (STVGP) using just the latitude, longitude, and the date from the training labels and ground measures could be an effective approach.[1] Unfortunately, the STVGP had a difficult time achieving good performance predicting future SWE measurements for locations or time periods outside the training sample. Also, given the size of the training dataset, the STVGP ran into memory constraints when trying to train the entire dataset on GPU. So we quickly moved in a different direction.

Consistent with the data-centric approach to machine learning, we turned our focus to exploring the possible data sources and feature engineering aspects of the problem.[2] The initial datasets we explored were tabular in nature, so we chose a gradient boosting decision tree (GBM) regressor which can capture non-linear effects between features and easily handle missing data values. The LightGBM regressor was chosen as the initial base model due to its ability to achieve high accuracy with faster training iterations than similar GBM implementations.[3] The faster training time provided a feedback loop for rapid feature prototyping and exploration.

Once the feature engineering was honed, we opted to train and ensemble three separate implementations of GBM to help avoid overfitting and capture the strengths of each implementation. In addition to LightGBM, we trained an XGBoost[4] model and a CatBoost[5] model. The XGBoost model cannot easily handle categorical features without special encoding, so it was trained on only the numerical features. The Catboost model specializes in using categorical features and the LightGBM model also has the ability to handle categorical features without encoding.

In order to determine how much weight to allocate to each model in the ensemble we used a generalized linear model (GLM) to obtain the optimal coefficients for each model in each region (Sierras, Central Rockies, Other). First, we obtained predictions on the Test dataset for each model. Then we subset the Test dataset by region. Next we fit a GLM on each region subset using the SWE predictions from each GBM implementation as independent variables and SWE as the dependent variable. To avoid overfitting to the Test dataset the coefficients were normalized to sum to 1. In the ensemble, the normalized coefficients provide the weight for each GBM implementation prediction in each region (Table 1, Fig. 1).

Data sources

We grouped the data sources into two categories, static and time-sensitive. Static data sources do not vary between prediction windows and typically represent geographic features of the locations. The time-varying data sources capture SWE features that vary for a particular location throughout the snow season.

The static data sources we used were the coordinates (latitude and longitude) from the ground measures and train/test labels, the digital surface data from the Copernicus Digital Elevation Model (DEM)[6], the

Climate Research Data Package (CRDP) Land Cover Gridded Map[7], and the CRDP Water Bodies Map. The three time-varying data sources used were the the ground measures features, the Modis Terra[8] and Aqua[9] Snow Cover Daily, and the NOAA High-Resolution Rapid Refresh (HRRR) Climate Data[10].

We also thoroughly considered and explored the Sentinel 1 Terrain Corrected Data[11] and the Landsat 8 Collection 2 Level-2 Data[12] before ultimately deciding not to use them in the final model.

Synthetic aperture radar (SAR) measurements from the Sentinel 1 can capture measurements day and night, independent of clouds or atmospheric conditions. The ability of SAR to capture data when Modis and other optical sensor reflectance data is obfuscated could provide valuable information to the model. However, one of the major downsides to SAR data is that terrain has a strong effect on the measurements. The different incident angles of measurement based on satellite location at the time of measurement combined with the mountainous terrain of the train/test locations results in inconsistent measurements. We attempted correcting for incidence angle and normalizing the data for particular locations, but the model performed better without including the Sentinel 1 SAR data.

The data from Landsat 8 Collection 2 Level-2 is promising because it offers a similar product to Modis but with greater resolution Modis (Landsat 30 m to Modis 500 m). The downside of the Landsat data is that it is collected less frequently than the Modis data. Since our modeling method of choice (GBM) requires tabular data inputs we relied on averaging NDSI values within a grid cell, thus losing the advantage of greater resolution. The Landsat data may be useful in models that can take advantage of the greater resolution, like convolutional neural networks. In the end, the Landsat data was captured too infrequently and considered redundant with the Modis data, so we did not include it in the final model.

Feature engineering

To maximize the amount of data used to train the models we created features using both the train labels and the ground measures. One data engineering challenge was to create homogenous features for the ground measures, which are 1 dimensional point measurements, and the grid cells, which are 2 dimensional GeoJson[13] polygons.

In the final model we only included features that improved the models' RMSE during cross-validation, using a permutation analysis that measured performance with and without the feature.

Static Features

The features created from the static data sources included:

- GeoJson grid cell file and ground measures features files
    - latitude, longitude
- Copernicus Digital Elevation Model (DEM)
    - elevation_m, elevation_var_m, south_elev_grad, east_elev_grad
- Climate Research Data Package (CRDP) Land Cover Gridded Map
    - lccs_0, lccs_1, lccs_2
- CRDP Water Bodies Map
    - water

Latitude and Longitude Features

The latitude and longitude were used as features in the model. In the case of the ground measures this was straightforward. For the grid cells the center point of the grid cell was used for the latitude and longitude feature, defined as the mean of the latitudes and the longitudes in the provided polygon.

Elevation Features

All of the data sources used to create features were localized using a latitude and longitude value (point measures) or a polygon based on latitude and longitude values (grid cells). The elevation features captured from the DEM demonstrate the differences in acquiring data for point measures and grid cells. The DEM captures elevation measurements at a 30 meter resolution.

The elevation_m feature represents the elevation in meters. For a point measure the elevation value can be directly queried from the DEM for the 30 meter grid corresponding to that point. To create the elevation_m feature for a 1 km^2 grid cell requires calculating the mean of all elevation values in the grid cell. The elevation_var_m represents the variance of elevation values for the grid cell in the DEM. The elevation_var_m is not applicable to a point measurement.

The elevation gradient features were also calculated from the DEM (Fig. 2). The elevation gradients were captured along two axes, east-west and south-north. For each axes the 1st discrete difference[14] was calculated for each 30 meter elevation value in the 1 km^2 grid cell. The mean over all of these discrete differences represents the gradient along that axis for the grid cell. For the south_elev_grad feature a positive value represents terrain with a greater Southern exposure than Northern exposure. Similarly, the east_elev_grad feature has a positive value for terrain with a greater Eastern exposure than Western Exposure. Since the elevation gradient features are also applicable to point measures we added and subtracted an epsilon of 0.001 to the latitude and longitude of the point measures to create a grid cell on which to calculate the gradient features.

Land Cover Category and Water Body Features

The CRDP land cover map classifies land surface into 22 classes at a 300 meter spatial resolution (Table 2). The `lccs_0` feature represents the most common land cover category for a grid cell or the land cover category for a point measure. The `lccs_1` and `lccs_2` represent the second and third most common land cover category for a grid cell, if applicable. Even though this is a categorical feature the models that use it in the ensemble, LightGBM and CatBoost, do not require a special encoding.

The CRDP water bodies map classifies areas into land and water at a 150 meter resolution. This data source is used to calculate the `water` feature which is a number between 0 and 1 and represents the percentage of the grid cell covered by a water body. This feature is not applicable to point measures which are assumed to be on land.

Time-Varying Features

Modis Snow Cover Daily Terra and Aqua Features

Several of the most important features were created from the Modis Satellite Snow Cover Daily product. One of the valuable aspects of the Modis NDSI data is it is captured daily from two different sets of satellites. The high-frequency of data capture for Modis allows for the creation of rolling averages, which smooth inherent noise in the measurements, within a time window that is close enough to the prediction date to be relevant to SWE prediction. Since the resolution for the Modis Snow Cover data is 500 meters,

there are typically 4-5 relevant measurements for each 1 km^2 grid cell and these were averaged for each grid cell. Next, rolling averages for the Modis Snow Cover Daily data (Terra and Aqua) were compiled with a 5-day and 15-day window from the SWE measurement date, creating the following features: NDSI_Snow_Cover_Terra_5_day, NDSI_Snow_Cover_Aqua_5_day, NDSI_Snow_Cover_Terra_15_day, NDSI_Snow_Cover_Aqua_5_day.

We experimented with different windows for the rolling averages (3, 5, 7, 15, 30) and in the end chose 5 and 15 because that minimized the loss in the models. With hindsight, a 7 day window may be preferable to the 5 day window for near real-time prediction because the Modis data is often available with a 2 day lag and with a 7 day window it is less important if the rolling average is missing values corresponding to the final day prior to SWE prediction.

NOAA HRRR Climate Data Features

The NOAA HRRR Climate data is captured at a 3 km resolution, so there is usually only one applicable value of each variable for each train/test sample regardless of whether it is a grid cell or a point measure (Fig. 3). A 3-day rolling average, counting back from the SWE measurement date, is compiled for the following NOAA HRRR Climate data fields:

- TMP_3_day - temperature [K]
- SPFH_3_day - specific humidity [kg/kg]
- PRES_3_day - pressure [Pa]
- PWAT_3_day - precipitable water [kg/m^2]

The value on the SWE measurement date is used for the following NOAA HRRR Climate data fields:

- WEASD - water equivalent of accumulated snow depth [kg/m^2]
- SNOD - snow depth [m]
- SNOWC - snow cover [%]

Snow Season Day Feature

The snow season day feature is calculated as a shift on the day of the year beginning at 0 (Dec. 1) incrementing by 1 each day, and reaching a maximum of 211 on the last SWE measurement day (June 30). Since 2020 was a leap year, 0 corresponds to November 30 and 212 to June 30.

Neighbor Relative SWE Feature

The neighbor relative SWE (NRSWE) feature is meant to represent the relative amount of SWE for the neighboring ground measurements of a grid cell, compared to historical values for the snow season period. The following steps show how the NRSWE feature is calculated:

1. Calculate 14 periods based on the snow season day value to get snow season period (Table 5).
2. Calculate the historical mean and standard deviation or each ground measurement/snow season period pair.
3. Use the historical mean and standard deviation to calculate the relative SWE, representing the Z-score (Equation 1) of each ground measurement.
4. For each grid cell find the 15 nearest neighbors and the distance (Equation 2) to each neighbor based on latitude and longitude in the ground measurement dataset.

5. Multiply the weighted inverse distance (Equation 3) for each of the 15 neighbors by each neighbor's relative SWE value (Equation 4).

The NRSWE feature captures the simple idea that when nearby ground measurement stations are measuring greater SWE than normal, it is likely that the location itself will have a greater SWE than normal. The effects of the NRSWE feature are highly localized by geography (Fig. 4, Fig. 5, Fig. 6, Fig. 7).

Explainability

SHAP values use a game theoretic approach to represent a feature's responsibility for a change in the model output.[15] Many of the features used in the GBM models are highly correlated. While decision trees are immune to multicollinearity, methods used to interpret decision tree based models, such as SHAP values[16], may suffer when features are correlated. We plot SHAP values to untangle the correlations between variables and learn which features provide the most information to the swe prediction model. We calculated SHAP values for the most accurate GBM implementation in the ensemble model, LightGBM.

According to SHAP values, the top 5 most important features in the LightGBM model are `NDSI_Snow_Cover_terra_15_day`, `elevation_m`, `WEASD`, `latitude`, and `snow_season_day`, in that order (Fig. 8). Interestingly, each region expressed a slightly different preference for features (Fig. 9, Fig. 10, Fig. 11). For example, `WEASD` is the most important feature in the Other region but is only the 4th most important feature in the Central Rockies region. In all regions the land cover category features (`lccs_0`, `lccs_1`, `lccs_2`), `water`, and `SNOWC` were the least important, but as noted above they did improve model scores during cross-validation, even if only marginally.

Another insight offered by the SHAP values feature importance plots is the directionality of features, whether an increase in the feature value leads to a higher or lower SWE prediction. One example of this is that the longitude feature shows a clear relationship between a low longitude (further East) and high SWE value in the Sierras region (Fig. 9). In the Central Rockies, the relationship between longitude and SWE value is not linear (Fig. 10), representing a more complex relationship along the East-West axis.

Interpreting feature importance directionality can also be helpful for checking intuitions. The elevation gradient features are important for both grid cells and point measurements. They may represent the intensity and duration of sunlight for a location and how well a location can hold snow. We calculate elevation gradients along two axes, East-West and South-North. It makes sense for there to be a linear relationship between the `south_elev_grad` feature because during the snow season mountain faces with Southern exposure are subject to direct sunlight and thus more snow melt. We do not expect this same relationship in the `east_elev_grad` feature and the SHAP plots reaffirm our intuition (Fig. 12, Fig. 13).

Another important step in model interpretation is to examine the impact that combinations of features have on the model's predictions. For example, the `water` feature results in a lower SWE prediction and the magnitude of this effect increases for greater values of `NDSI_Snow_Cover_aqua_15_day` or `NDSI_Snow_Cover_terra_15_day` (Fig. 15). This can be viewed as the model correcting for over predictions of SWE when the Modis Satellite NDSI value is high due to water reflectance in the location.

The importance of the `NDSI_Snow_Cover_terra_15_day` feature is heavily entangled with spatial and temporal features, like latitude, elevation_m, and snow_season day. According to SHAP values, the `NDSI_Snow_Cover_terra_15_day` feature has a bigger impact on SWE predictions at higher latitudes (Fig. 16). While the importance of the `snow_season_day` feature varies between regions, the unifying pattern between all regions is that high values for `NDSI_Snow_Cover_terra_15_day` in the middle of the

season correspond to `snow_season_day` having greater positive impact on SWE prediction, with the inverse relationship at the beginning and end of the snow season (Fig. 17). The details of these plots may be useful to scientists seeking to understand how snow seasons vary between regions.

The `elevation_m` feature relates to linear increases in the SWE prediction up until approximately 3,600 meters, at which point it levels off and begins to decrease (Fig. 18). Why the SWE prediction value begins to plateau or decrease above 3,600 meters in the Central Rockies and Sierras is a question worthy of further research. The `NDSI_Snow_Cover_terra_15_day` feature is correlated with an increase in the slope describing the linear relationship between `elevation_m` and its impact on SWE prediction (Fig. 18). That is, SWE prediction is more sensitive to the `elevation_m` feature for higher values of `NDSI_Snow_Cover_terra_15_day`. It is possible that the model is using the `elevation_m` value as a counterweight to `NDSI_Snow_Cover_terra_15_day` for SWE prediction.

One feature that may be of particular interest to scientists if the climate of the Western United States changes in the coming years is the effect of the `TMP_3_day` feature on SWE prediction. There is a non-linear relationship between temperature and SWE (Fig. 19). Temperature is correlated with humidity, temperatures significantly below freezing (273.15 K) result in dry conditions that aren't suitable for SWE formation and temperatures significantly above freezing are too warm for SWE accumulation. The ideal temperature for SWE formation appears to be 265-280 K. Rising temperatures could also result in a shorter snow season (Fig. 20).

Model robustness

Training, validation, and testing

We only used temporal stratification to separate the observations and relied on the features of the model to distinguish between samples with different geography, elevation, or snow conditions. Observations were divided into a Train dataset that contained all train labels and ground measures from 2013 to 2019 and a Test dataset that contained labels and ground measures from 2020 and 2021.

We ran 5-fold cross-validation on the Train dataset to tune the hyperparameters of the GBM models and perform feature selection. The metric used to evaluate model performance was the root mean squared error (RMSE). After selecting features and optimal hyperparameters we performed SWE prediction on the Test dataset to evaluate the model's ability to generalize to unseen data.

For all three GBM implementations the majority of the reduction in RMSE came in the first 100 iterations of training, but each model continued to improve RMSE on the train and test dataset for many subsequent iterations as shown in the learning curves (Fig. 21). The LightGBM model achieved the best performance on the Test dataset, followed by the XGBoost model and then the Catboost model (Table 3, Fig. 22). The results in Table 3 demonstrate that the ensemble of GBM models generalized better than any individual model and capitalized on the relative strength of each GBM implementation in certain regions.

Performance across conditions

The overall low RMSE across the entire Test dataset demonstrates that the model is able to generalize to a variety of conditions across time, SWE levels, and location. The model produced the lowest RMSE in the Other region, and the highest RMSE in the Sierras (Table 3). This is especially significant because the Test dataset is skewed towards samples in the Sierras (Table 4), so achieving a better performance in the Sierras would have an outsized effect on the overall RMSE. While a detailed analysis identified several

areas where the model produced less accurate SWE predictions it can be difficult to pinpoint the exact reason for SWE miscalculation due to the high correlation between variables.

For instance, SWE levels are highly correlated with time (snow season period) in all locations, even if the peak SWE period differs by region. The RMSE for each model peaked between snow_season_period 5 and 10 (Fig. 23). This also corresponds to the snow_season_periods with the highest SWE values (Fig. 24, Fig. 25).

Another technique we used to determine which features were tied to varying model performance was to fit a GLM with the scaled features as the independent variables and the ensemble models RMSE as the dependent variable (Table 6). The magnitude of the coefficients of the scaled features represent a linear relationship between those variables and RMSE. We plotted features (`elevation_m`, `WEASD`) that had large coefficients against the corresponding RMSE for each location. The plots demonstrate that the model tended to have less accurate SWE predictions for lower values of `WEASD`, higher values of `elevation_m`, and SWE values between 20 and 70 (Fig. 26, Fig. 27, Fig. 28).

Edge cases/missing values

One of the virtues of many GBM implementations is graceful handling of missing values by default. LightGBM handles missing numerical values during training by assigning them to the split in the decision tree that most reduces the loss.[17] During prediction LightGBM follows the pattern of assigning missing values used in training or falls back on reasonable heuristics. While the details may differ, XGBoost and CatBoost are also capable of handling missing values.

In a near real-time prediction scenario time-sensitive data may not always be accessible in a timely manner. Given that the GBM implementations can handle missing data values, the ensemble model is able to create predictions even with missing data. But how does it perform? One of the characteristics of our ensemble model that makes it more robust for near real-time prediction is that it relies on multiple data sources to inform the time-sensitive features used as a signal for SWE prediction. There are three independent time-sensitive data sources: Modis NDSI Snow Cover, NOAA HRRR Climate Data, and SNOTEL/CDEC ground measures that inform the `neighbor_relative_swe` feature. If one or two of these sources is unavailable the models' predictions are still better than if all three were unavailable (Fig. 29 & Table 7).

Having multiple related time-sensitive features can also help clear ambiguities. There are two sets of Modis satellites Terra and Aqua. If the NDSI Snow Cover features disagree between Terra and Aqua the model will give more importance to the WEASD feature (Fig. 30).

Despite GBMs being able to handle missing values, it is still preferable to impute a value if doing so would improve the loss of the model. As can be seen in Table 7 the model is most sensitive to missing Modis values. We took additional steps to reduce missing Modis values where possible. The Modis NDSI Snow Cover product will mask out values where the data quality is low, including if there is sufficient cloud cover.[18] However, there is a strong temporal correlation between NDSI Snow Cover values. Therefore, we created a rolling average of NDSI Snow Cover values that ignored missing values. We used a 15-day window to capture long term trends and a 5-day window to capture short term trends. This method not only reduced inherent noise in the Modis NDSI Snow Cover measurement but also greatly reduced missing values.

Considerations for use

Machine specifications

The code for this project was written in Google Colab notebooks with Google Drive as a backend. These freely available technologies allow for anyone with an internet connection to train and run the SWE prediction model. No GPU was necessary to train the model but acquiring training data can take a lot of time and computational resources. Therefore, it is recommended to run the Colab notebook with Background Execution enabled and a High-RAM runtime.

Once data has been collected, executing the model to produce predictions is fast. The main bottle-neck for real-time prediction is waiting for the time-sensitive data source to update. The most important data source, Modis, may have up to a 2 day lag. One strategy that would allow for model prediction to run sooner without sacrificing much accuracy is to use only Modis measurements available on the SWE prediction day allowing for the model predictions to be produced on the same day.

Recommendations

Given more time to work on the challenge of SWE prediction in the Western United States there are several data sources and methods that could improve the results. The training labels were derived from a combination of ground-based snow telemetry (ie. SNOTEL and CDEC sites) and airborne measurements (ie. ASO data). During training it would be useful to know which labels fall into each respective category, even if that information is not available during prediction time. If there are subtle differences in the measurements that impact model performance then it might inform model architecture.

As described above, with the satellite based imaging data there is a trade off between the higher time frequency, lower spatial resolution of Modis and the lower time frequency, higher spatial resolution of Landsat. In the end, the greater frequency of Modis was more valuable to our model than the greater resolution of Landsat. However, one compelling idea for future work is to combine datasets with greater resolution (Landsat, Sentinel, DEM) with a modeling approach that could take advantage of the greater resolution, like a convolutional neural network. This could allow for more granular SWE predictions with higher resolution than 1 km^2 which may be more accurate if the data can support it.

Another idea for future work is to implement features that capture temporal changes in time-sensitive data, rather than only performing rolling averages. One example is that atmospheric pressure is highly correlated with elevation (higher elevations have lower pressure). So the addition of the feature representing atmospheric pressure (`PRES_3_day`) in our model did not add much information that wasn't already present in the `elevation_m` feature. However, a feature that represents change in atmospheric pressure over some period of time (i.e. 3-day window) might add relevant information to the model about weather patterns that lead to SWE, and this information should not be as highly correlated with elevation.

A final thought, SWE is a cumulative measure. Modeling changes in SWE throughout the snow season may shed light on the dynamics of SWE accumulation and the conditions that give rise to SWE. It would also be interesting to connect SWE prediction to metrics about SWE runoff and the effects on river conditions and water availability beyond just the snow season.

Appendix A: Tables and Figures

Region	LightGBM	XGBoost	CatBoost
Sierras	0.349412	0.212484	0.438103
Other	0.872737	0.007129	0.120133
Central Rockies	0.646812	0.181185	0.172002
All	0.567160	0.167770	0.265070

Table 1: Optimal weight of each model by Region

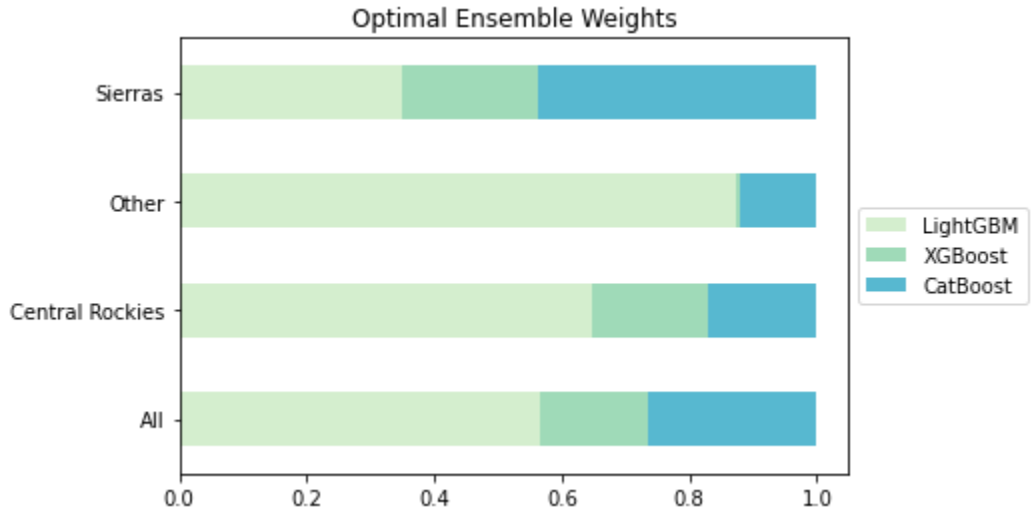

Figure 1: Optimal weight of each model by Region

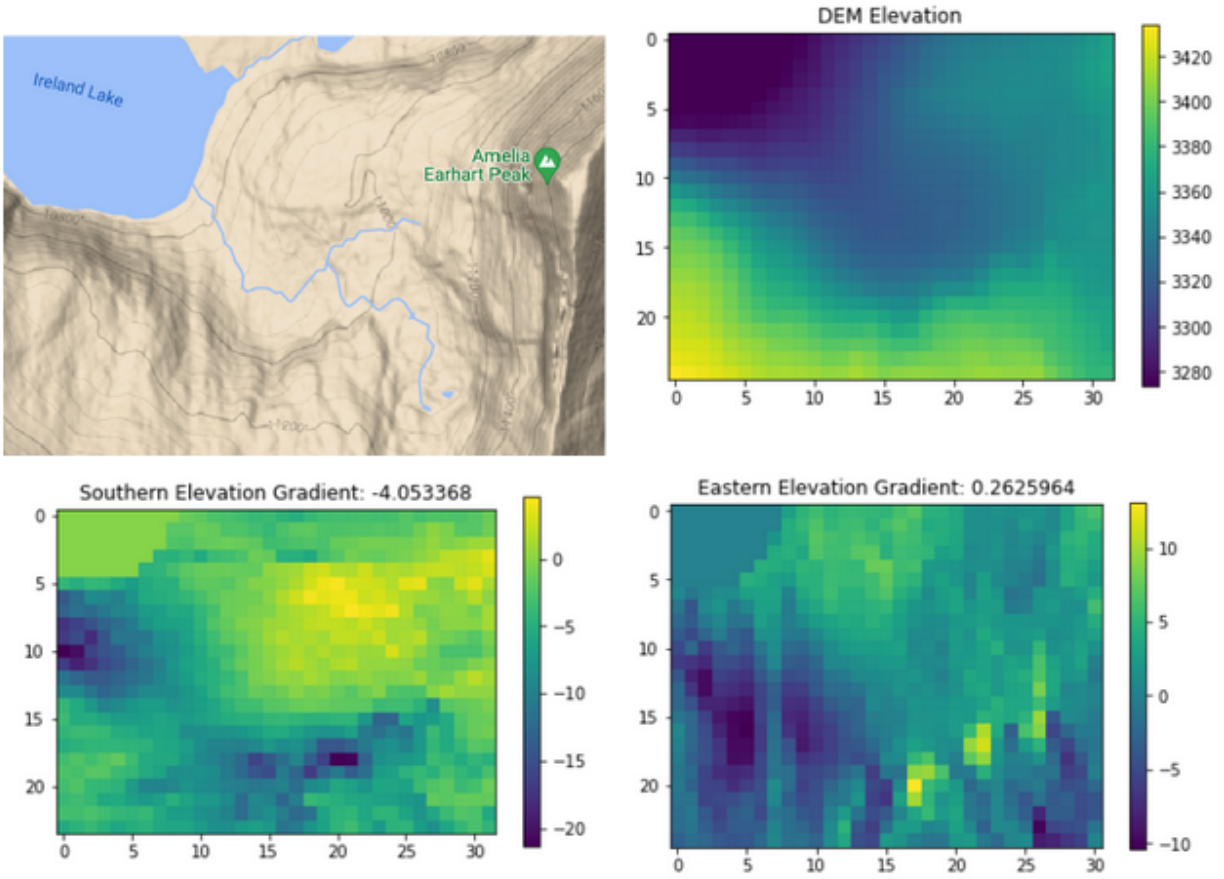

Figure 2: Illustrative example of elevation features for a location in the Sierra region with mean latitude 37.783973 and mean longitude -119.298367. Top right: terrain view of grid cell from Google Maps. Top left: raw pixel elevation pixel values from DEM. Bottom left: Southern elevation gradient pixel values. Bottom right: Eastern elevation gradient pixel values

Region	lccs_0	Label	Train data count
Sierras	70	Tree cover, needleleaved, evergreen, closed to open (>15%)	3123
Sierras	200	Bare areas	917
Sierras	120	Shrubland	503
Sierras	210	Water bodies	37
Central Rockies	70	Tree cover, needleleaved, evergreen, closed to open (>15%)	1565
Central Rockies	130	Grassland	534
Central Rockies	120	Shrubland	264
Central Rockies	200	Bare areas	250

Other	70	Tree cover, needleleaved, evergreen, closed to open (>15%)	2881
Other	200	Bare areas	106
Other	130	Grassland	103
Other	10	Cropland, rainfed	36

Table 2: Top 4 land cover categories by Region

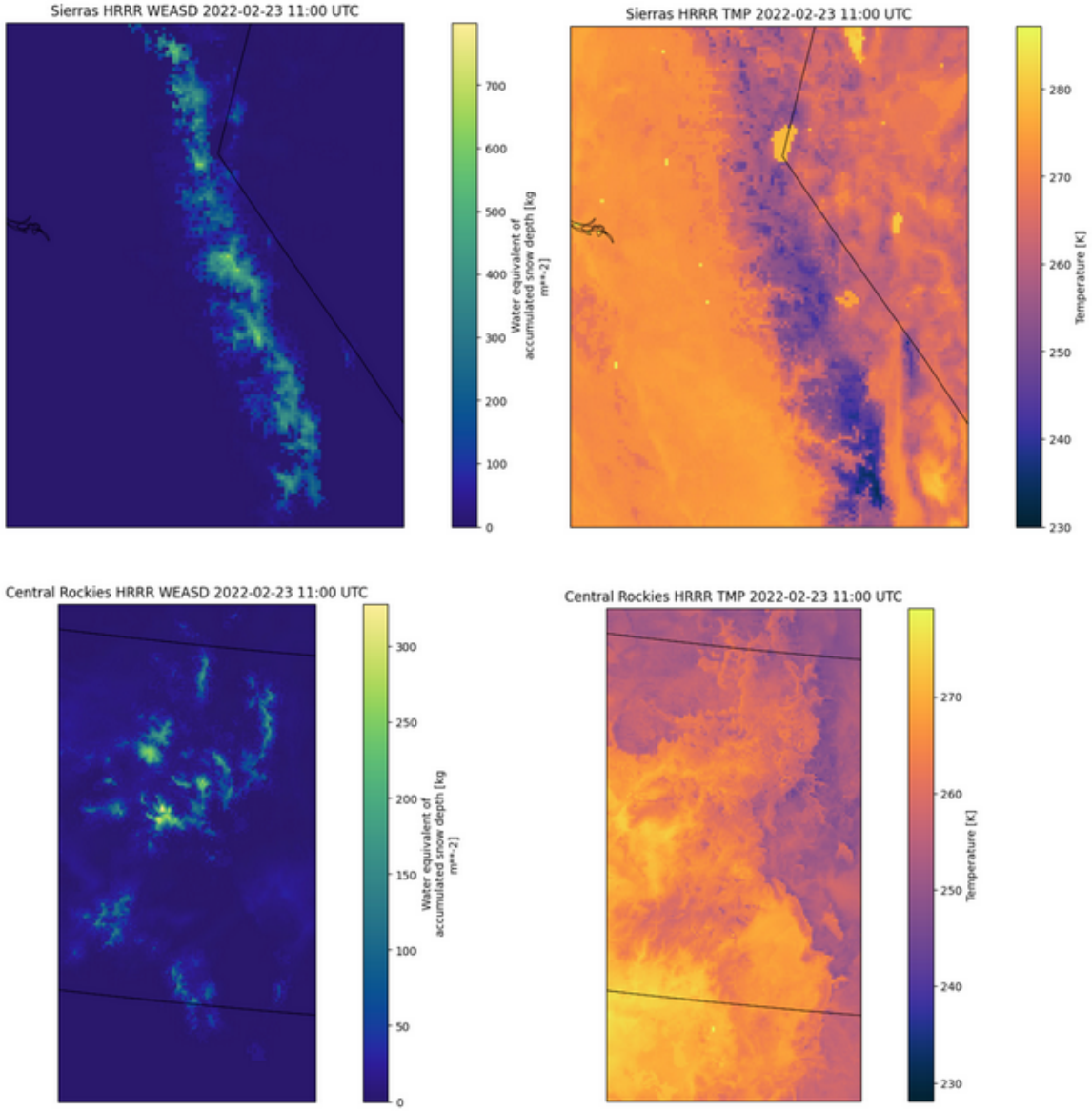

Figure 3: NOAA HRRR climate data as of 2022-02-23 11:00 UTC. Top left: water equivalent of accumulated snow depth for the Sierras. Top right: temperature for the Sierras. Bottom left: water equivalent of accumulated snow depth for the Central Rockies. Bottom right: temperature for the Central Rockies

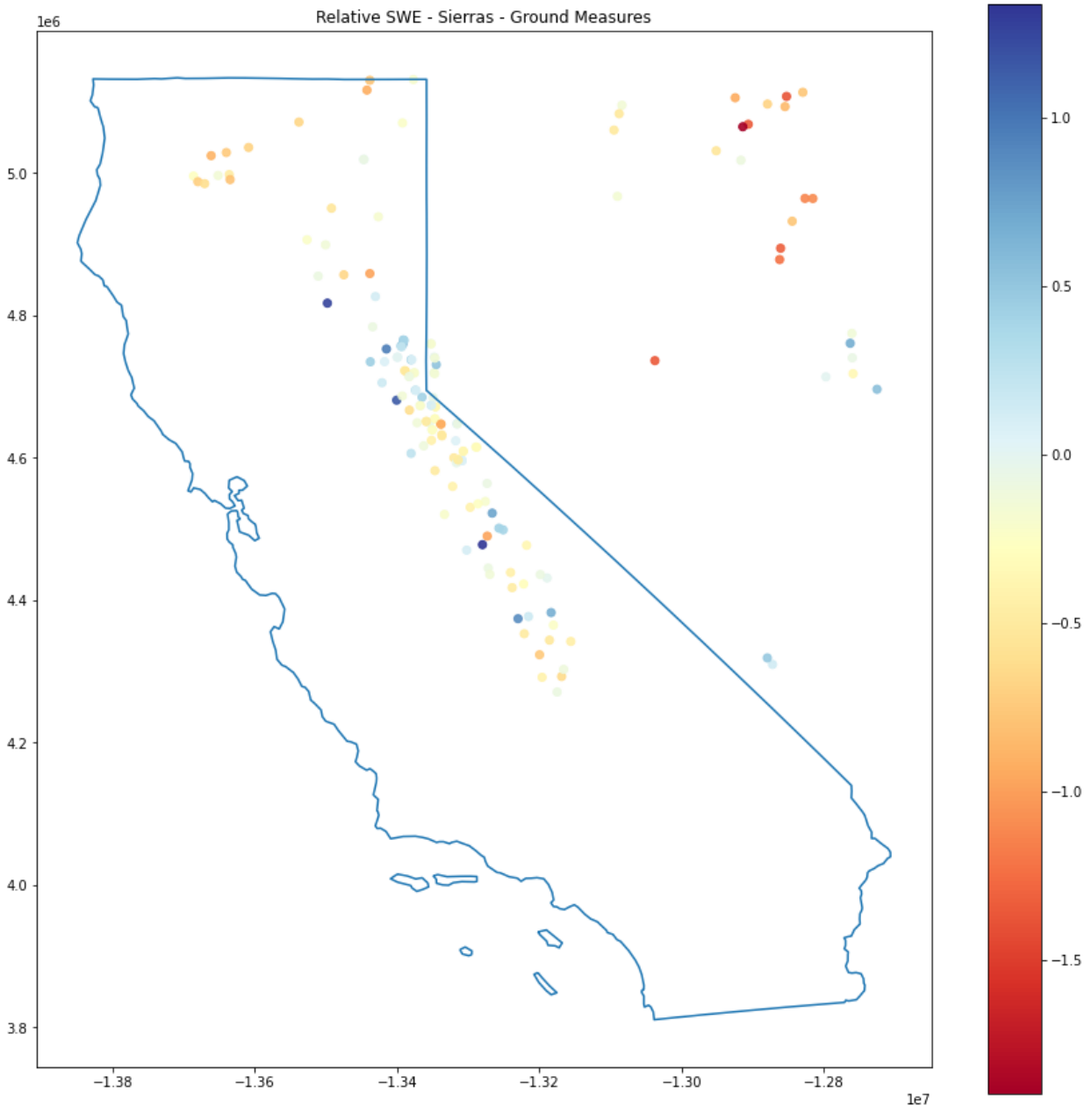

Figure 4: Sierras Region relative SWE of ground measures

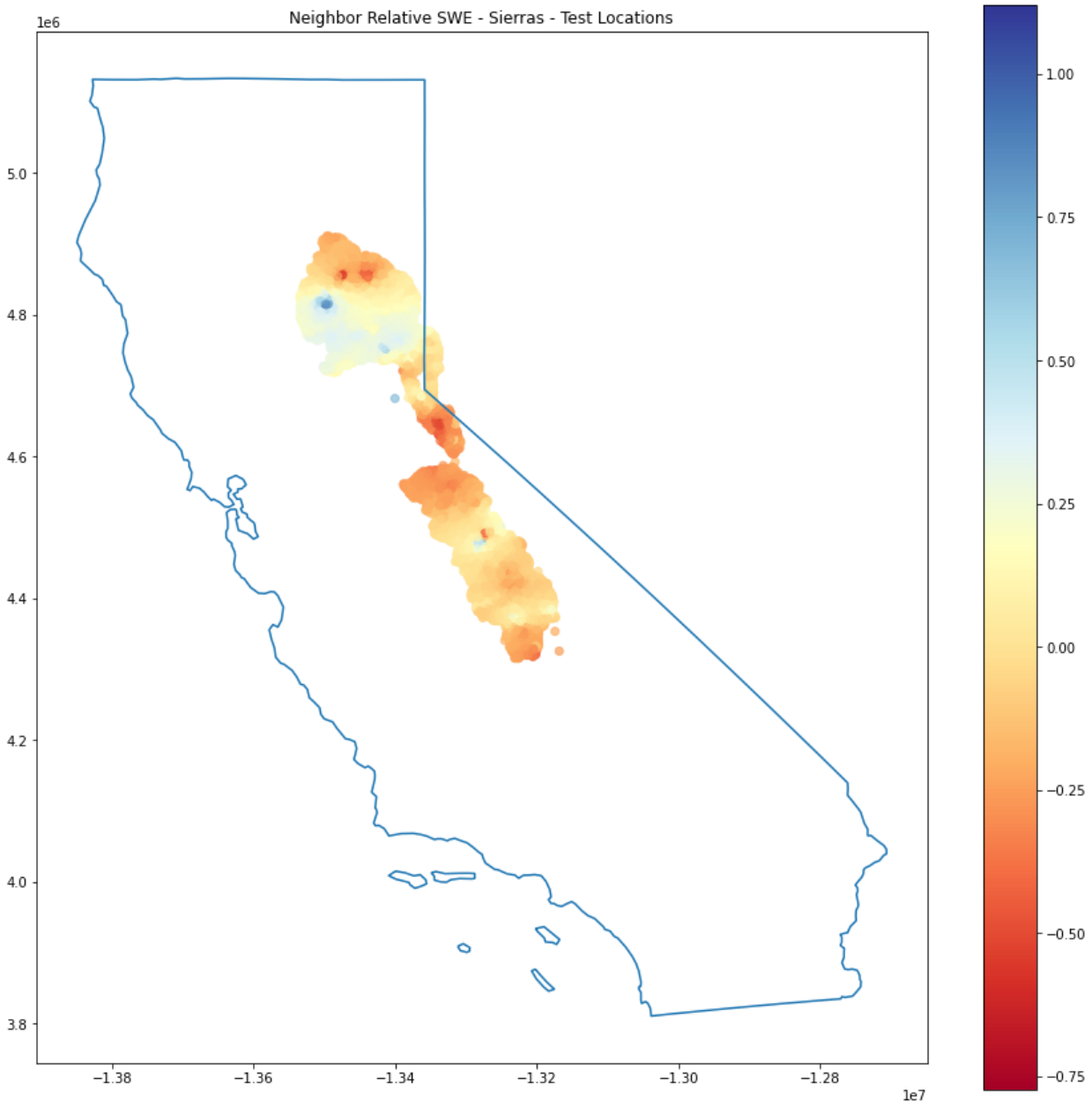

Figure 5: Sierras Region neighbor relative SWE of test locations

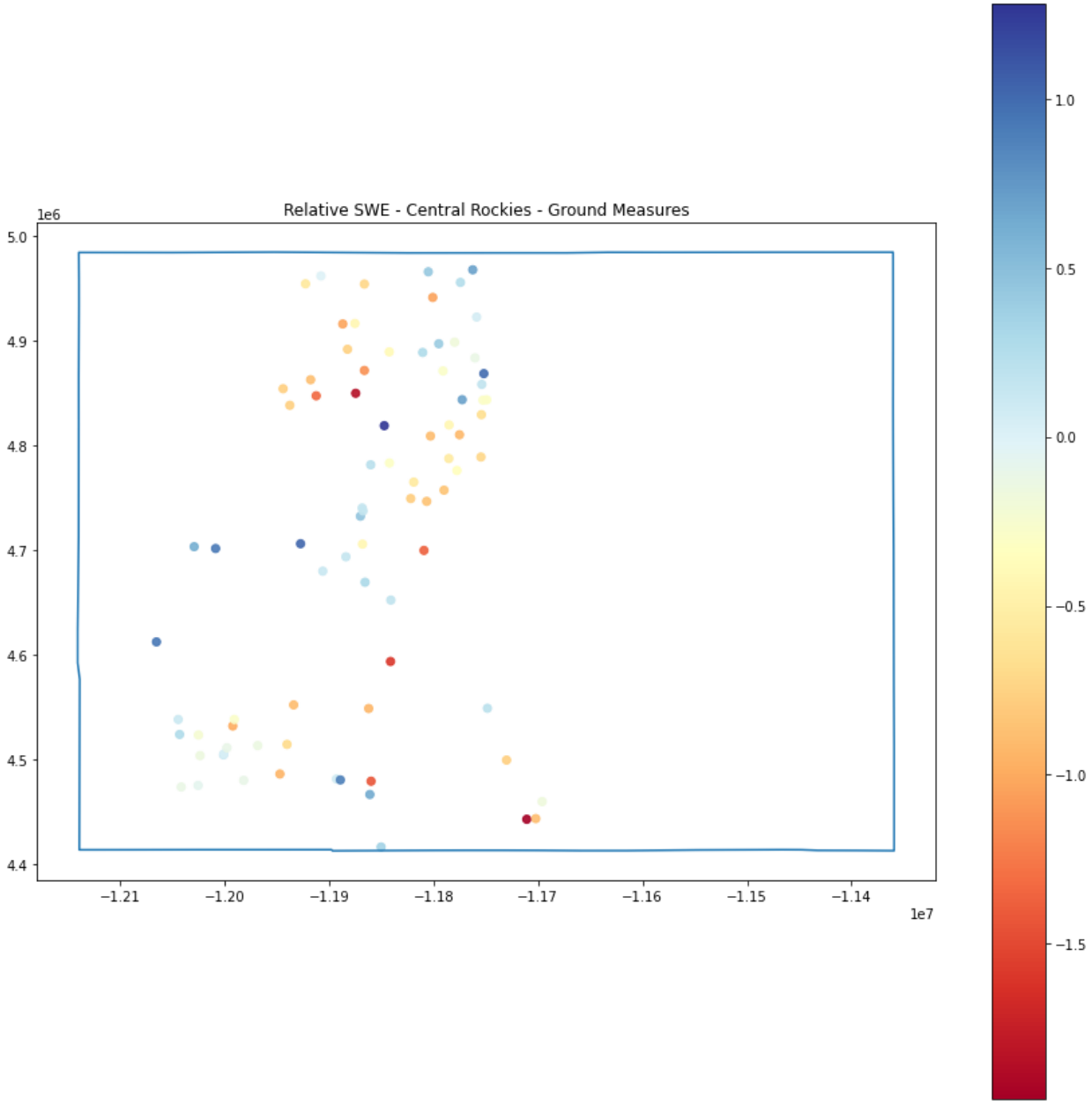

Figure 6: Central Rockies Region relative SWE of ground measures

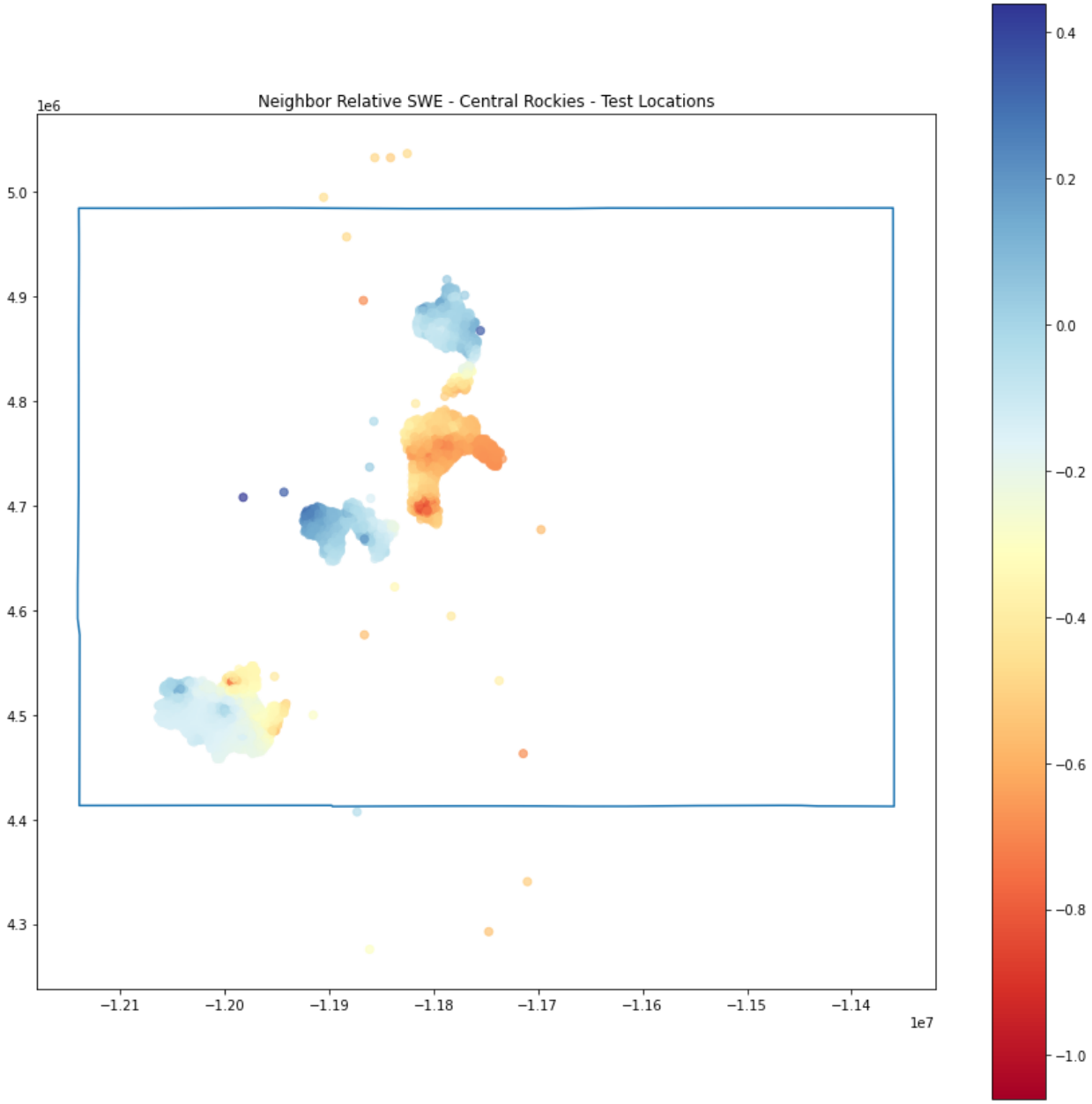

Figure 7: Central Rockies Region neighbor relative SWE of test locations

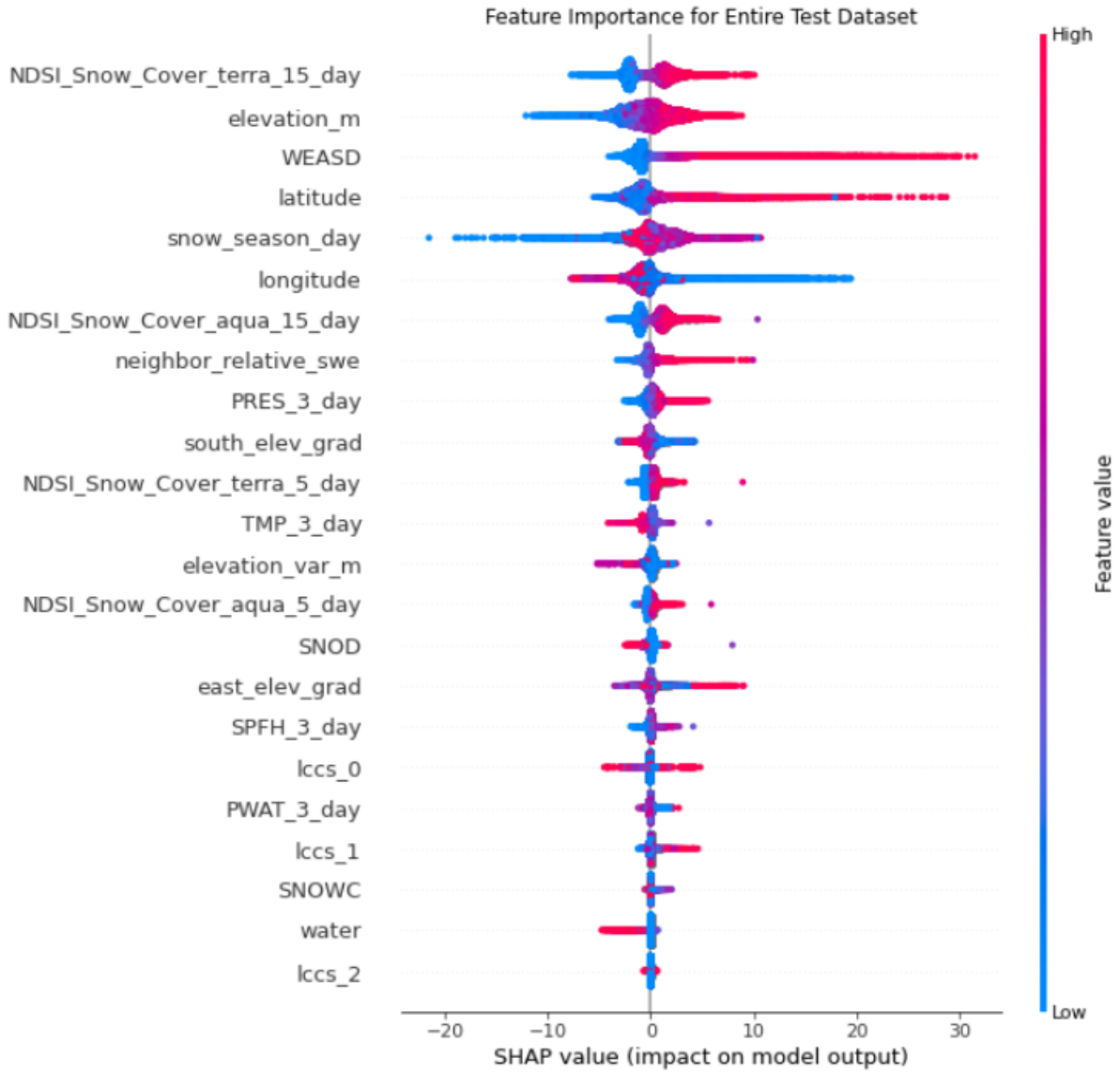

Figure 8: SHAP Value Summary Plot of feature importance for the entire test dataset in the LightGBM model

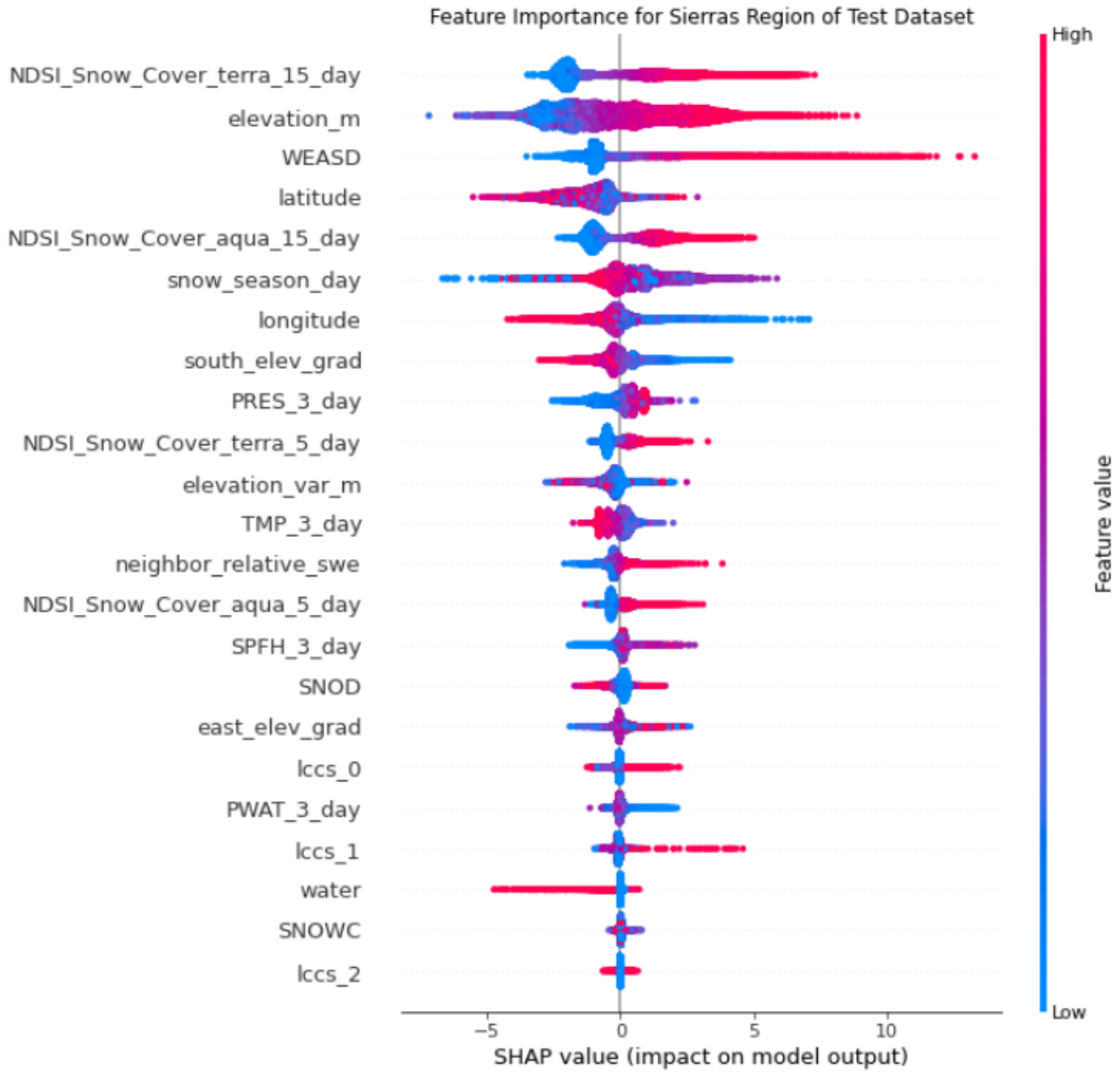

Figure 9: SHAP Value Summary Plot of feature importance for the Sierras Region of the test dataset in the LightGBM model

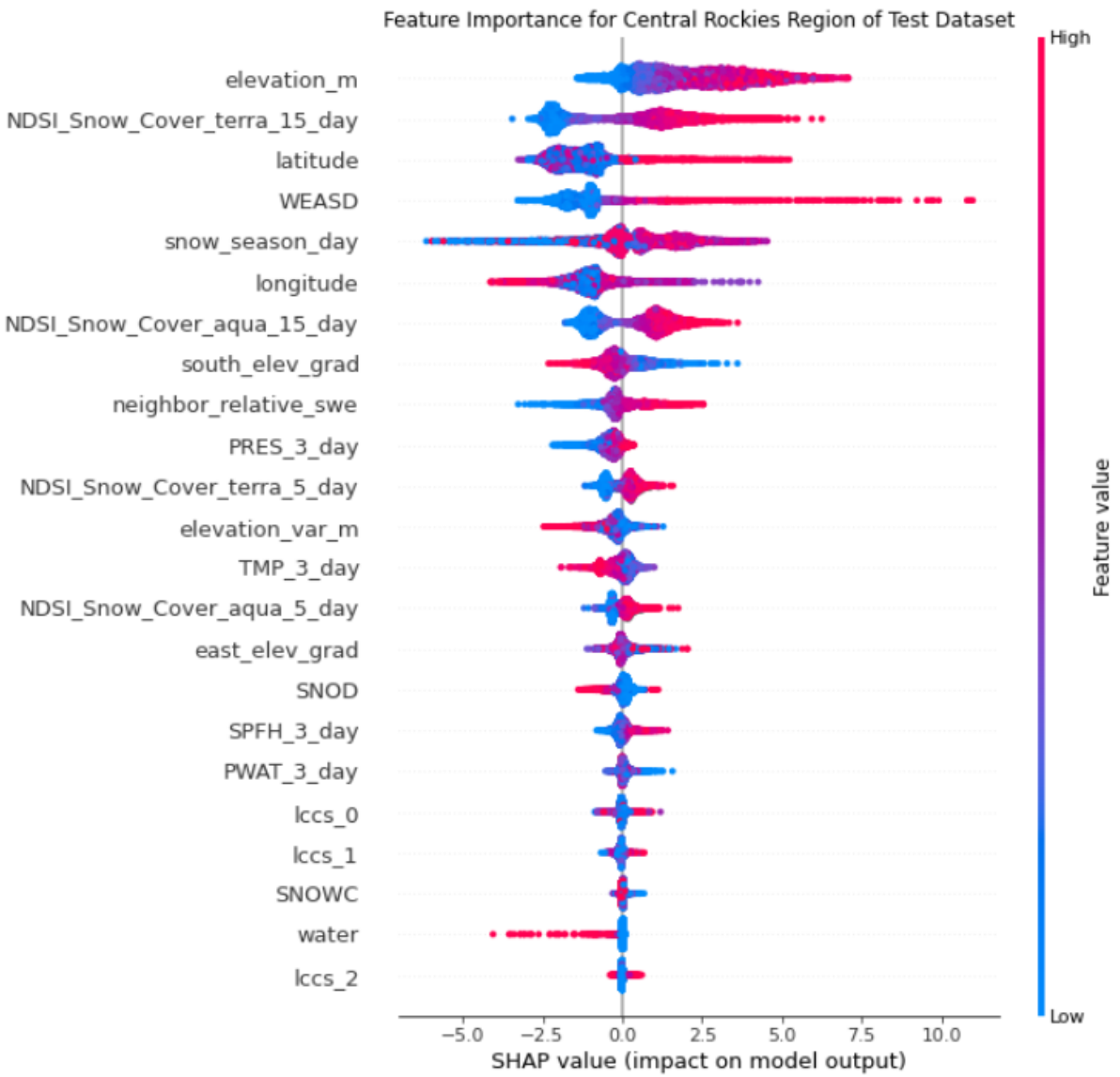

Figure 10: SHAP Value Summary Plot of feature importance for the Central Rockies Region of the test dataset in the LightGBM model

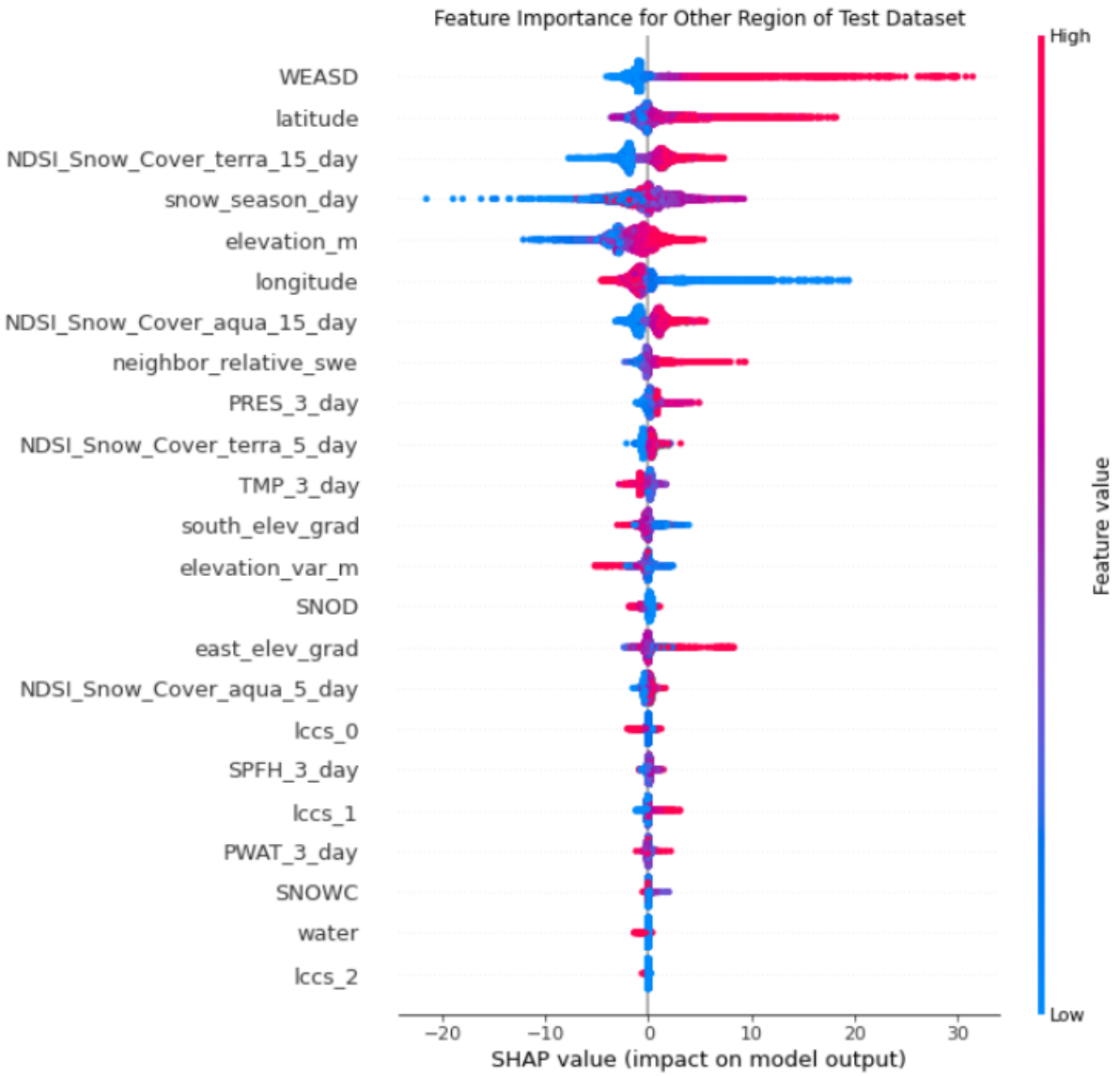

Figure 11: SHAP Value Summary Plot of feature importance for the Other Region of the test dataset in the LightGBM model

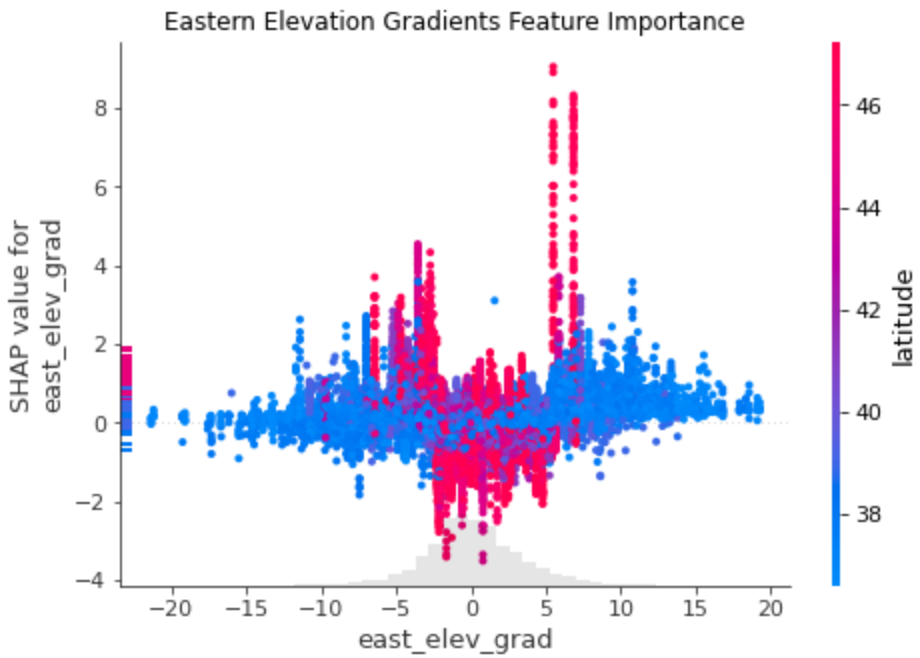

Figure 12: Plot of SHAP value for east_elev_grad feature against its shap_value colored by latitude. This plot demonstrates a symmetrical relationship for the importance of negative or positive values of east_elev_grad

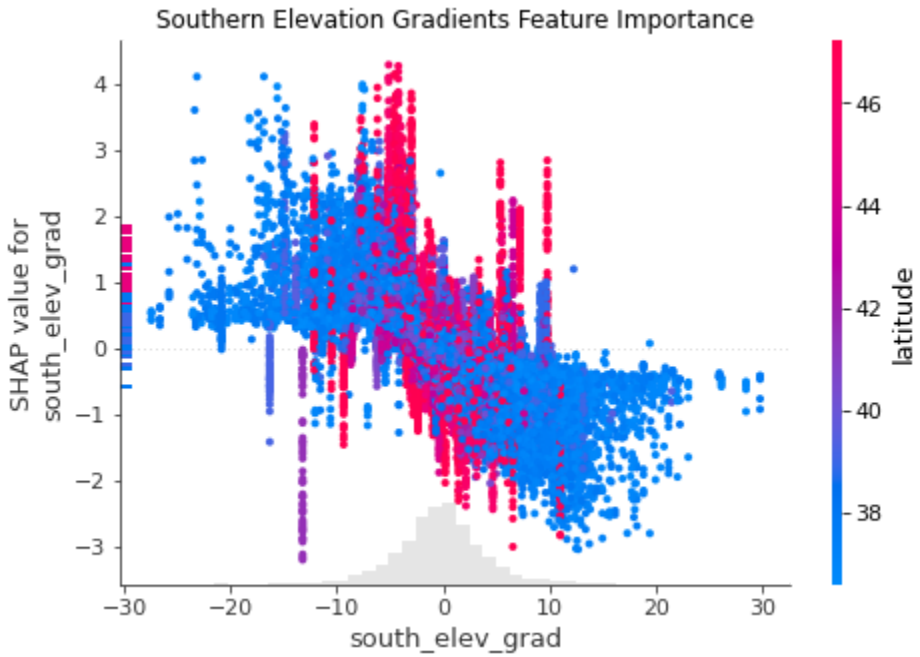

Figure 13: Plot of SHAP value for south_elev_grad feature against its shap_value colored by latitude. This plot demonstrates a relationship where negative values of south_elev_grad contribute to greater SWE prediction and the opposite for positive values of south_elev_grad.

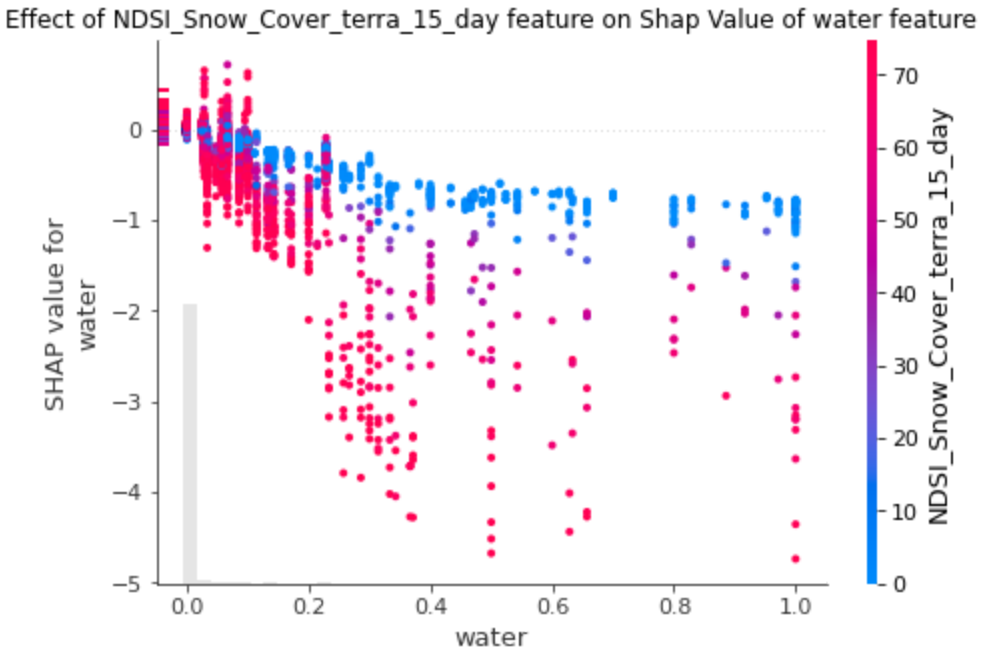

Figure 14: The water feature negatively impacts SWE prediction at greater values of NDSI_Snow_Cover_terra_15_day

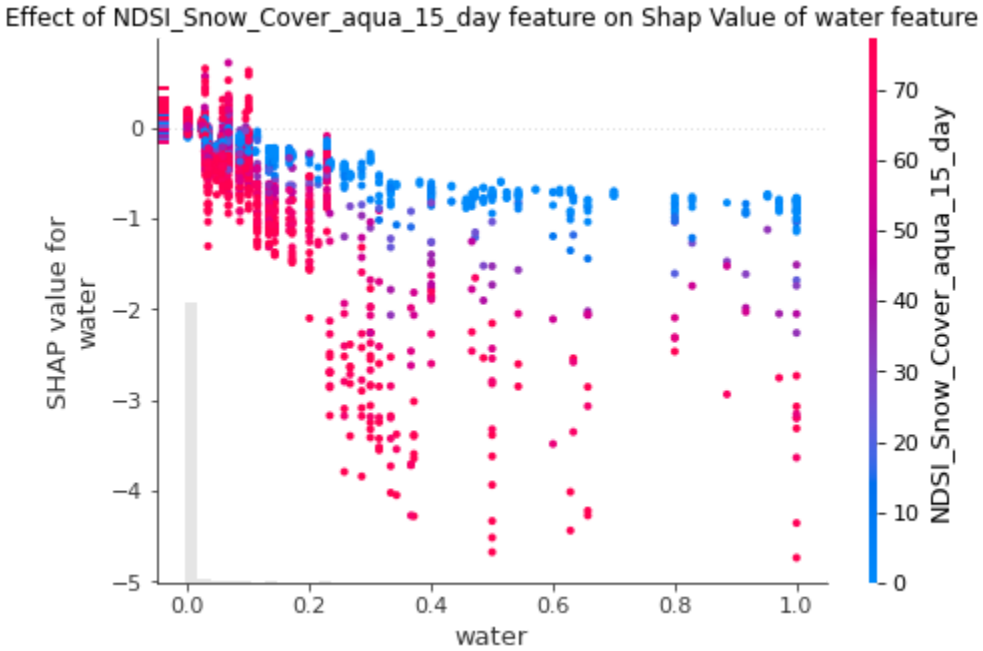

Figure 15: The water feature negatively impacts SWE prediction at greater values of NDSI_Snow_Cover_aqua_15_day

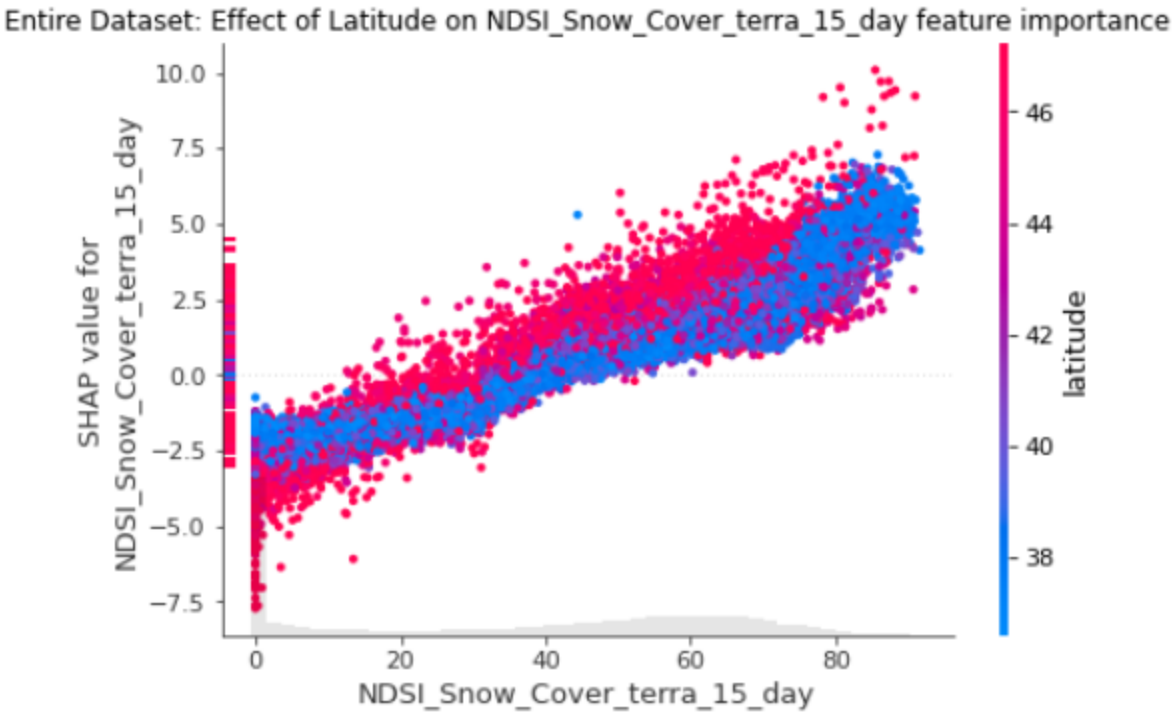

Figure 16: Plotting the values of the `NDSI_Snow_cover_terra_15_day` feature against its SHAP values and coloring the plot based on the `latitude` feature demonstrates the effect of geography on the model's use of `NDSI_Snow_cover_terra_15_day` in predicting SWE.

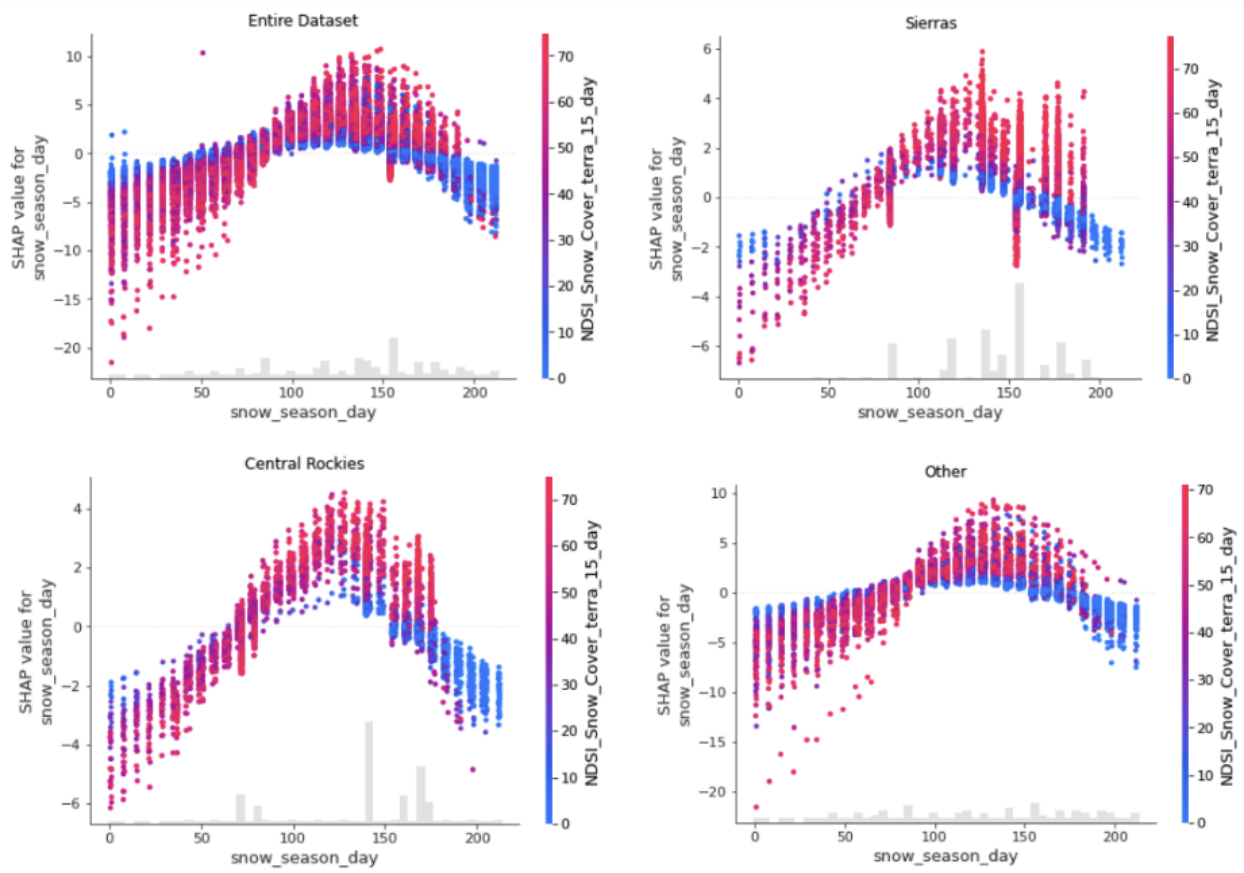

Figure 17: Plots for the entire dataset and each region showing the SHAP value for the snow_season_day feature colored by the value of NDSI_Snow_Cover_Terra_15_day

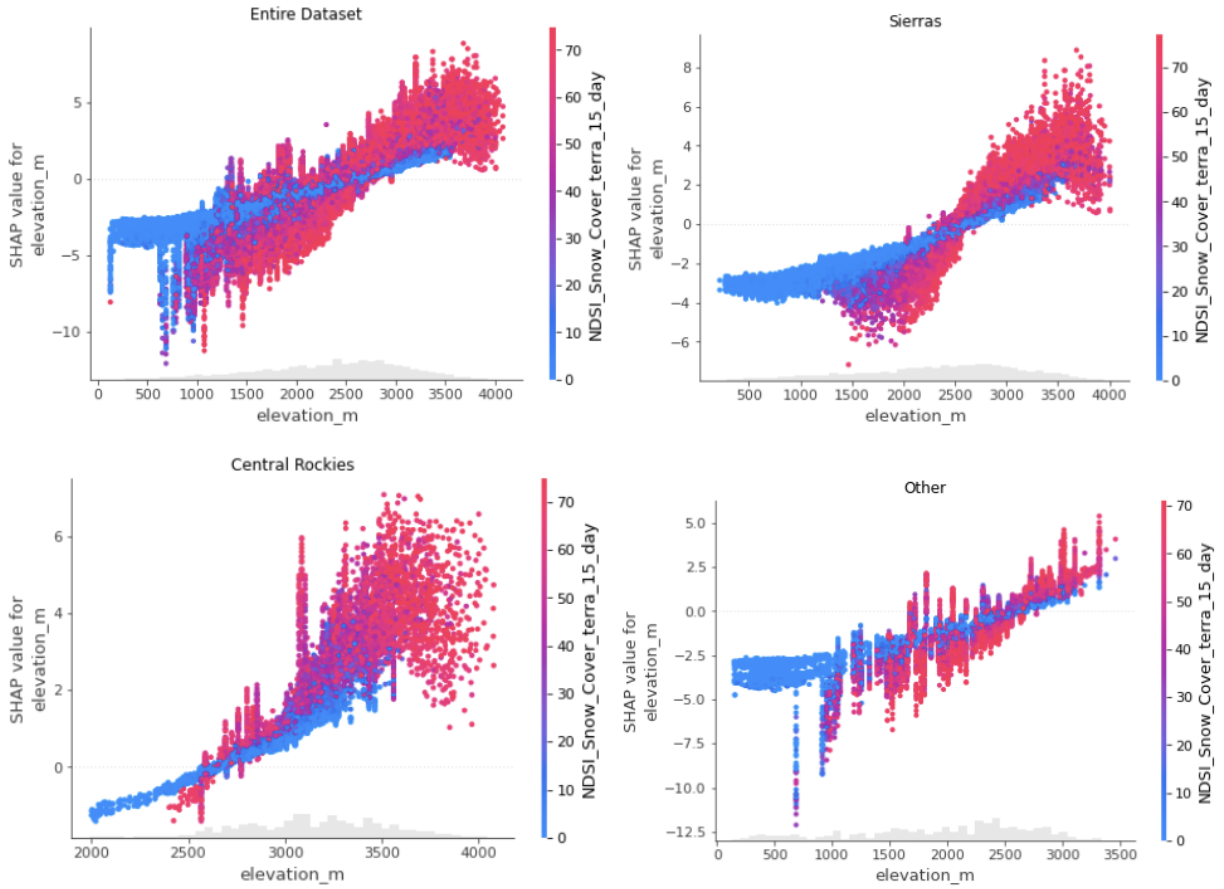

Figure 18: Plots for the entire dataset and each region showing the SHAP value for the elevation_m feature colored by the value of NDSI_Snow_Cover_Terra_15_day

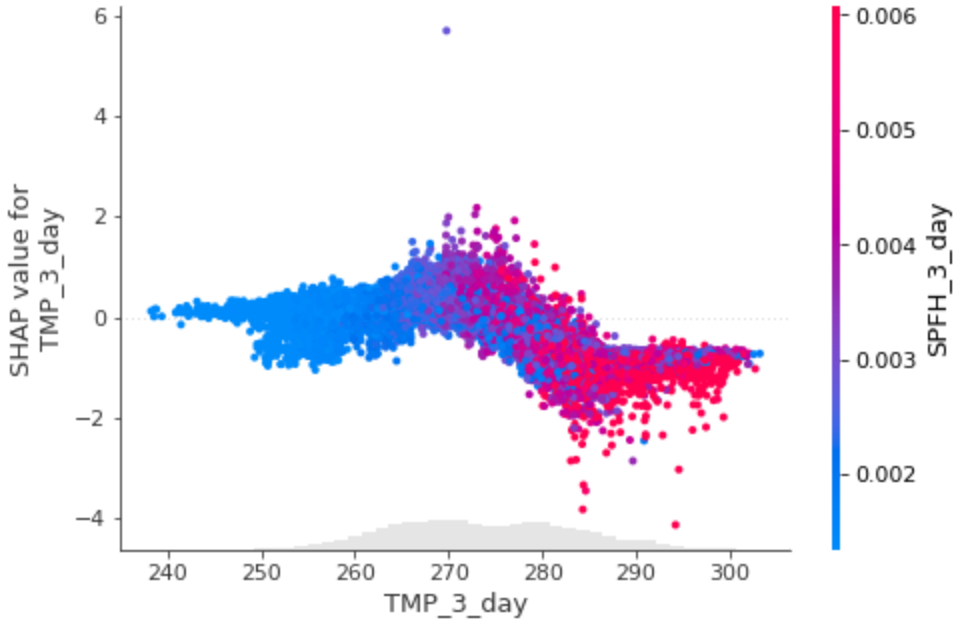

Figure 19: SHAP values for TMP_3_day feature against TMP_3_day values colored by SPFH_3_day demonstrates correlation between temperature and humidity and the impact on SWE.

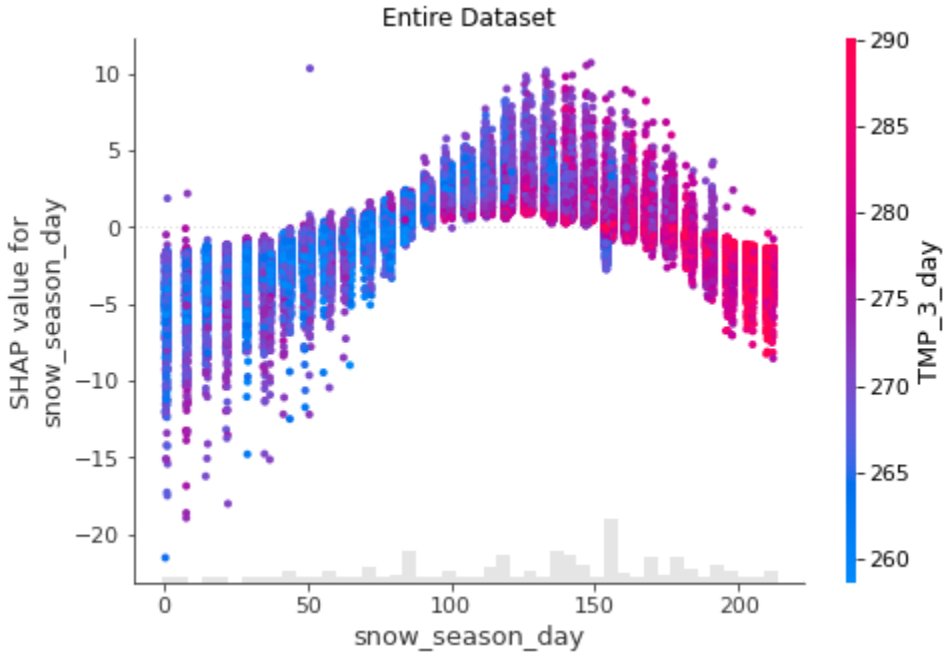

Figure 20: SHAP values for snow_season_day feature against snow_season_day values colored by TMP_3_day demonstrates the relationship between temperature and snow season and the impact on SWE.

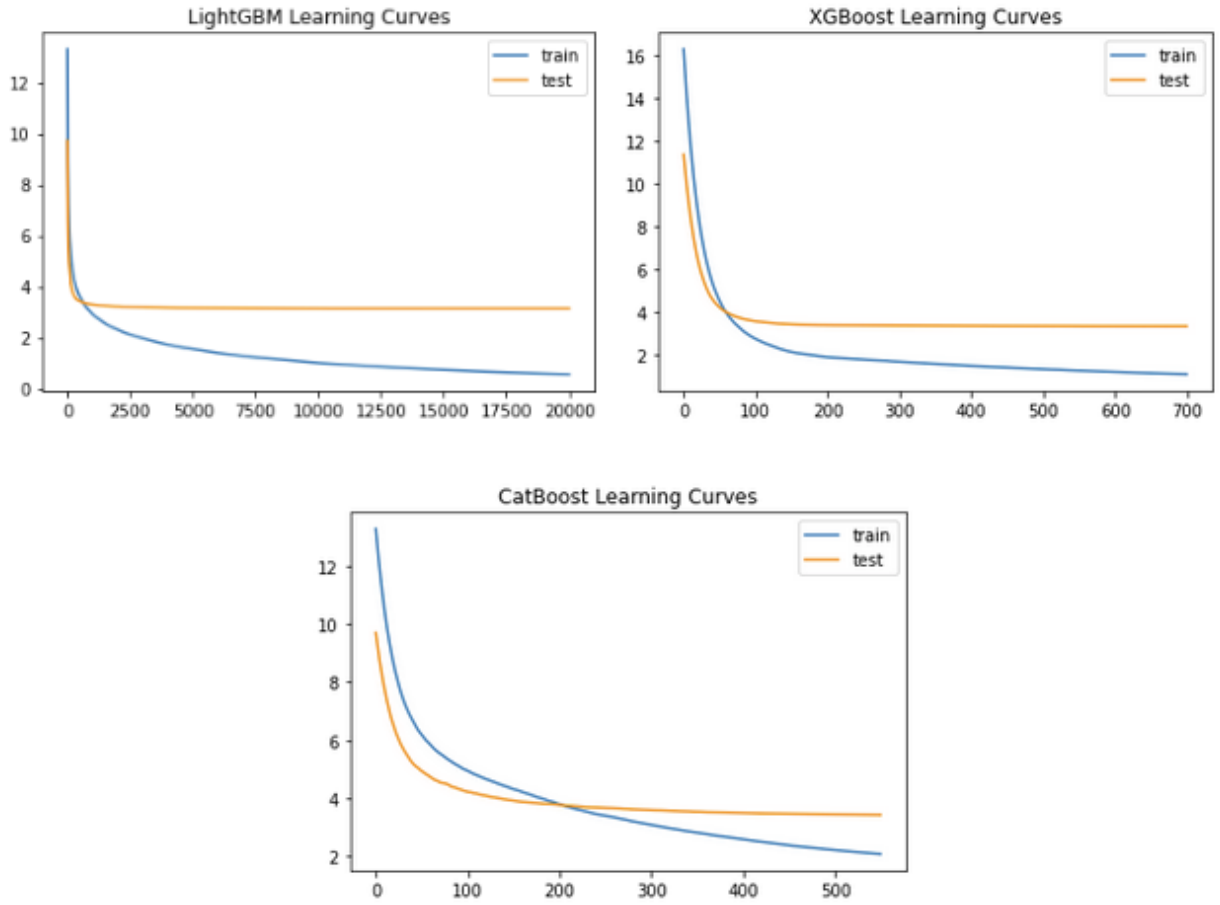

Figure 21: Train and test learning curves for each implementation of GBM. Top Left: LightGBM, Top Right: XGBoost, Bottom Middle: CatBoost

Region	ensemble_rmse	xgb_rmse	lgb_rmse	cb_rmse
Central Rockies	3.004530	3.188042	3.032472	3.240249
Other	2.481624	2.839434	2.451692	3.213350
Sierras	3.501435	3.858107	3.695853	3.714822
All	3.215590	3.546199	3.345466	3.534426

Table 3: Model performance by region and by GBM implementation

Figure 22: Model performance by region and by GBM implementation

Region	Test Data Count
Central Rockies	6646
Other	10715
Sierras	27880

Table 4: Count of Test data points by Region

	snow_season_day		date	
snow_season_period	**min**	**max**	**min**	**max**
0	1	8	2020-12-01	2020-12-08
1	15	22	2020-12-15	2020-12-22
2	29	37	2020-01-07	2020-12-29
3	44	51	2020-01-14	2020-01-21
4	58	65	2020-01-28	2020-02-04
5	72	79	2020-02-11	2020-02-18
6	86	93	2020-02-25	2020-03-03

7	100	107	2020-03-10	2020-03-17
8	114	121	2020-03-24	2020-03-31
9	128	135	2020-04-07	2020-04-14
10	142	149	2020-04-21	2020-04-28
11	156	163	2020-05-05	2020-05-12
12	170	177	2020-05-19	2020-05-26
13	184	191	2020-06-02	2020-06-09
14	198	212	2020-06-16	2020-06-30

Table 5: Mapping of snow_season_period to snow_season_day range and min and max dates in 2020.

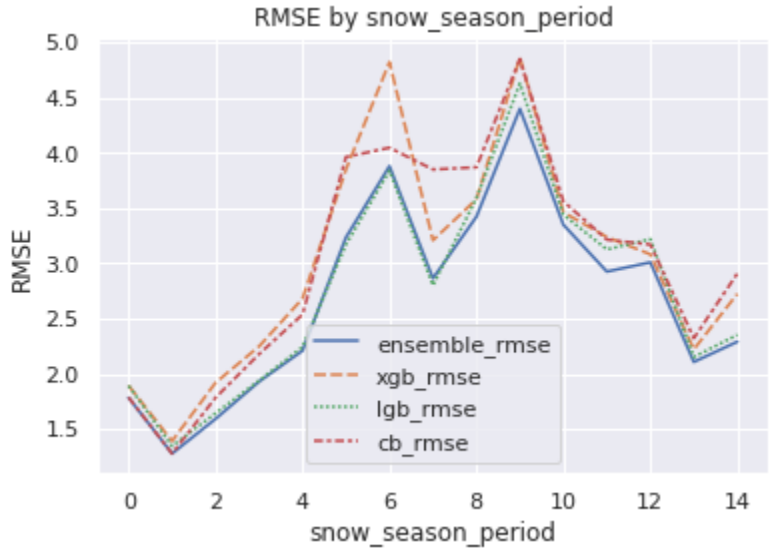

Figure 23: RMSE by snow_season_period for each implementation of GBM and the ensemble model.

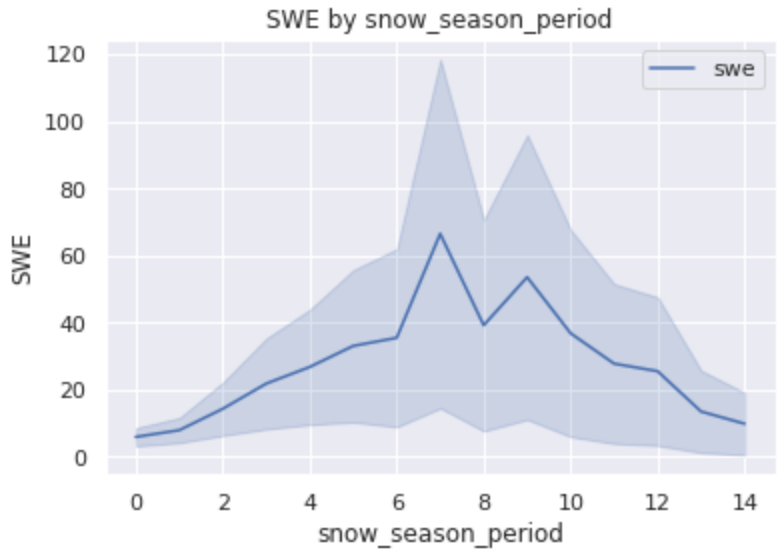

Figure 24: SWE by snow_season_period for the entire dataset

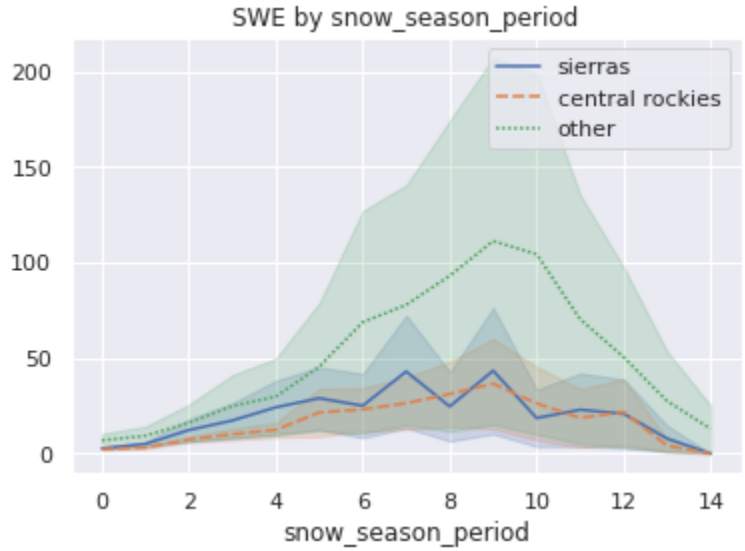

Figure 25: SWE by snow_season_period for each region

Feature	RMSE Prediction Coefficient
elevation_m	9.957913
PRES	9.190137
WEASD	7.148038
swe	4.506799

NDSI_Snow_Cover_terra_5_day	2.921632
NDSI_Snow_Cover_aqua_15_day	2.826167
snow_season_day	2.658220
NDSI_Snow_Cover_terra_15_day	2.121431
elevation_var_m	1.430602
SPFH	1.236600
south_elev_pct	0.527718
REFC	0.491895
east_elev_pct	0.355194
water	0.274641
neighbor_relative_swe	0.026504
NDSI_Snow_Cover_aqua_5_day	0.016759
PWAT	-0.203423
east_elev_grad	-1.052932
south_elev_grad	-1.113634
latitude	-1.138722
TMP	-1.561560
SNOWC	-1.713549
longitude	-2.011902
SNOD	-4.879448

Table 6: Coefficients of a linear model fitting scaled features to the ensemble model RMSE to demonstrate which features tended to vary with the RMSE.

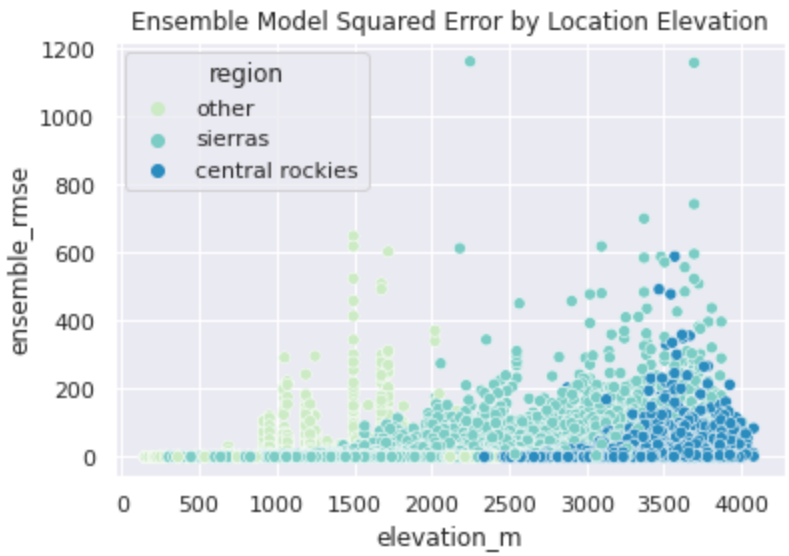

Figure 26: Ensemble model squared error by location elevation in meters colored by region.

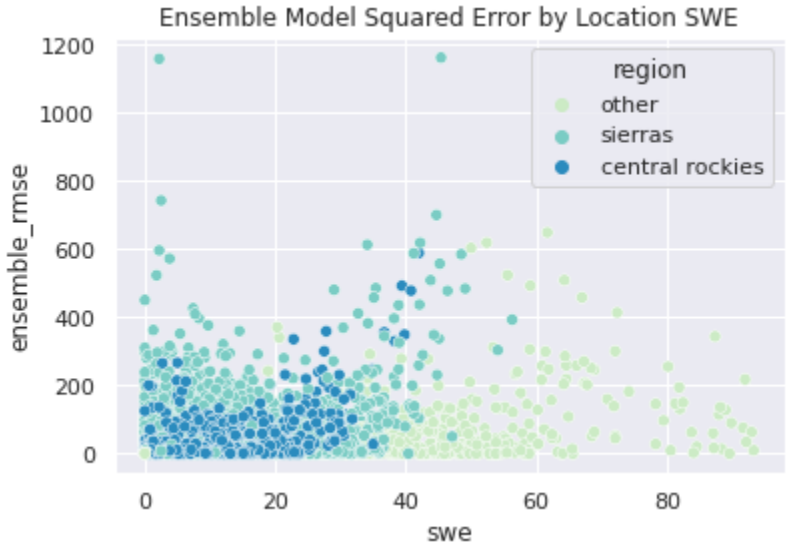

Figure 27: Ensemble model squared error by location SWE colored by region.

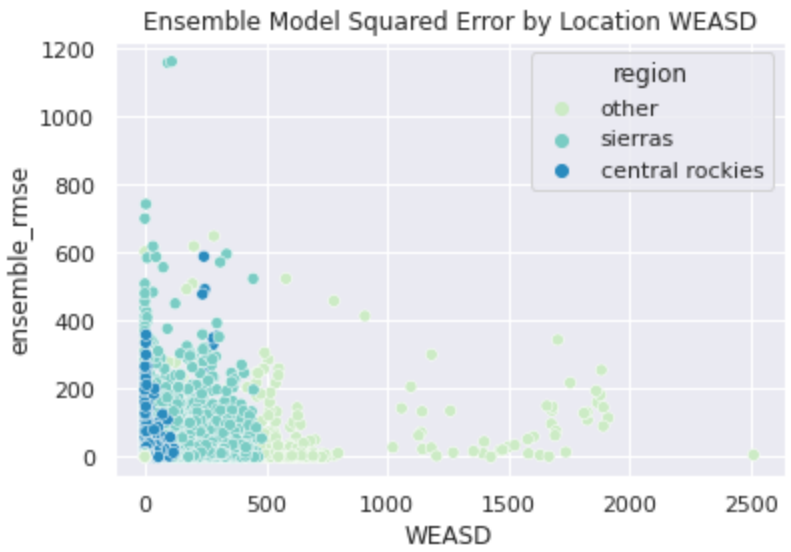

Figure 28: Ensemble model squared error by location WEASD colored by region.

Figure 29: Model performance on Test dataset when time-sensitive features have missing values.

Model	All Data	Missing Climate & Modis & NRSWE	Missing Modis	Missing Climate & NRSWE
ensemble_rmse	3.215589	4.863030	4.489747	3.386738

xgb_rmse	3.546199	5.753576	6.325957	4.701713
lgb_rmse	3.345465	5.353256	4.435824	4.258091
cb_rmse	3.534426	6.997055	5.779365	4.502428

Table 7: Model performance on Test dataset when time-sensitive features have missing values.

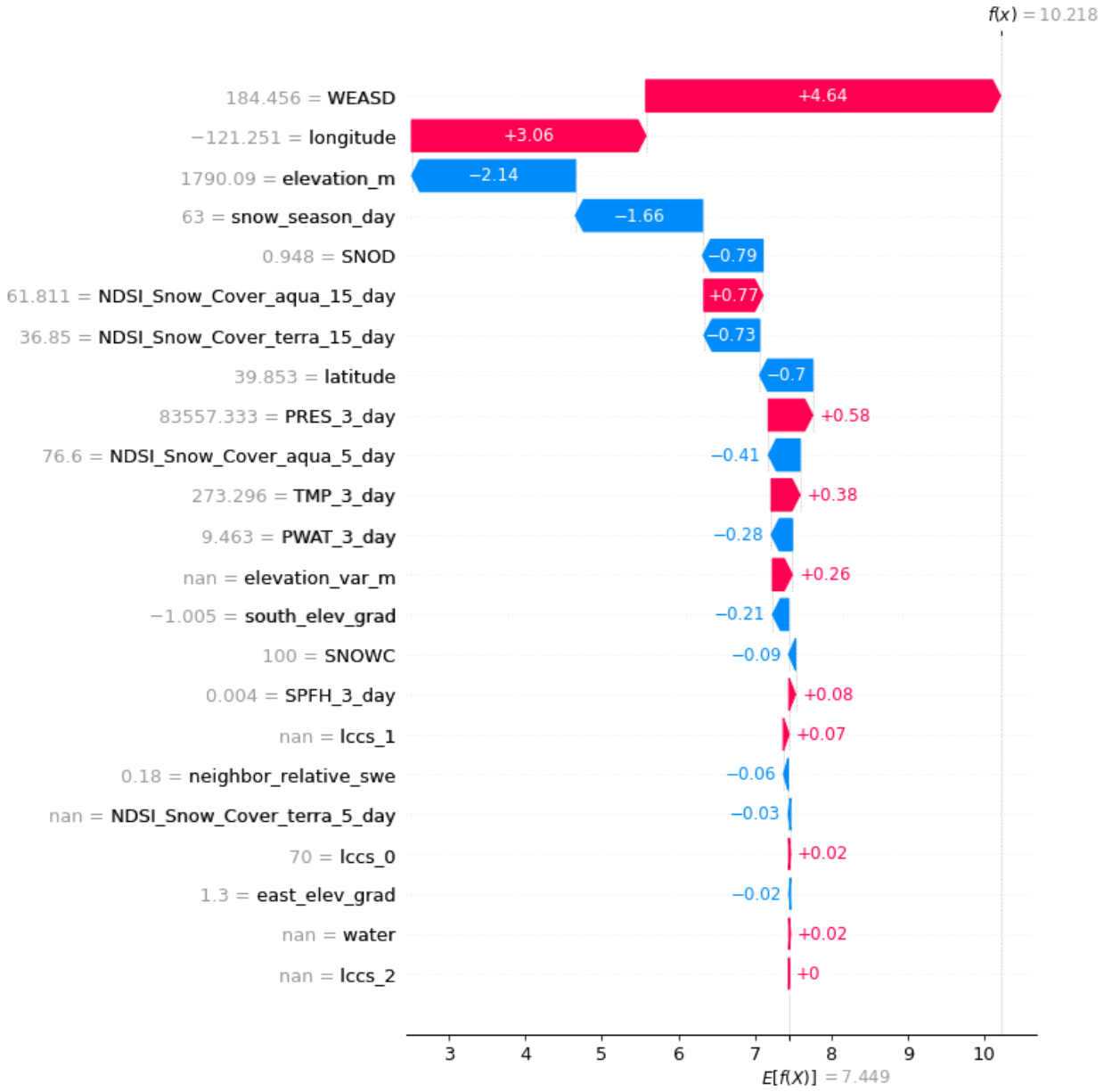

Figure 30: SHAP value waterfall plot that shows feature importance for a particular data point. In this case the values for NDSI_Snow_Cover_aqua_15_day and NDSI_Snow_Cover_terra_15_day disagree, so the model places more emphasis on the WEASD value.

Appendix B: Equations

$$z = \frac{\bar{X} - \mu}{\hat{\sigma}} \tag{1}$$

$$d_n = distance\ to\ n^{th}\ neighbor \tag{2}$$

$$p_n = \frac{d_n^{-1}}{\sum_{m=1}^{15} d_m^{-1}} \tag{3}$$

$$\sum_{n=1}^{15} z_n * p_n \tag{4}$$

Endnotes

1. https://proceedings.neurips.cc/paper/2021/file/c6b8c8d762da15fa8dbbdfb6baf9e260-Paper.pdf
2. https://datacentricai.org/
3. https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf
4. https://xgboost.readthedocs.io/en/stable/
5. https://catboost.ai/
6. https://object.cloud.sdsc.edu/v1/AUTH_opentopography/www/metadata/Copernicus_metadata.pdf
7. http://maps.elie.ucl.ac.be/CCI/viewer/download.php#ftp_dwl
8. https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MOD10A1
9. https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MYD10A1
10. https://www.nco.ncep.noaa.gov/pmb/products/hrrr/
11. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S1_GRD#description
12. https://www.usgs.gov/landsat-missions/landsat-collection-2-level-2-science-products
13. https://geojson.org/
14. https://numpy.org/doc/stable/reference/generated/numpy.diff.html
15. https://github.com/slundberg/SHAP
16. https://christophm.github.io/interpretable-ml-book/SHAPley.html#disadvantages-13
17. https://github.com/microsoft/LightGBM/issues/2921
18. https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MOD10A1#bands