Supplementary Materials for:

Effects of Bark Beetle Outbreaks on Forest Landscape Pattern in the Southern Rocky Mountains, U.S.A.

Kyle C. Rodman [1,*], Robert A. Andrus [2], Cori L. Butkiewicz [1], Teresa B. Chapman [3,4], Nathan S. Gill [5], Brian J. Harvey [6], Dominik Kulakowski [7], Niko J. Tutland [8], Thomas T. Veblen [4] and Sarah J. Hart [1,8]

[1] Department of Forest and Wildlife Ecology, University of Wisconsin, 1630 Linden Dr., Madison, WI 53706, USA
[2] School of the Environment, Washington State University, P.O. Box 642812, Pullman, WA 99164, USA
[3] Colorado Field Office, The Nature Conservancy, 2424 Spruce St., Boulder, CO 80302, USA
[4] Department of Geography, University of Colorado, Guggenheim 110, 260 UCB, Boulder, CO 80309, USA
[5] Department of Natural Resources Management, Texas Tech University, Box 42125, Lubbock, TX 79409, USA
[6] School of Environmental and Forest Sciences, University of Washington, Box 352100, Seattle, WA 98195, USA
[7] Graduate School of Geography, Clark University, 950 Main St., Worcester, MA 01610, USA
[8] Department of Forest and Rangeland Stewardship, Colorado State University, 1472 Campus Delivery, Fort Collins, CO 80523, USA
* Correspondence: krodman2@wisc.edu

Table S1: Summary of US Forest Service Aerial Detection Surveys (ADS) from 1997 to 2019 [1] in the Southern Rocky Mountains, USA. "Mapped Area" gives the total area within ADS-mapped perimeters for each insect or pathogen (i.e., "Agent Code"). For this comparison, polygons for each agent code were dissolved across survey years, such that areas with overlapping polygons representing the same agent were only included once in area calculations. Agents with mapped areas < 100 km^2 were excluded. Note that surveyed polygons typically include some unaffected areas, and therefore the total mapped area typically overestimates the area of tree damage or mortality.

Agent Code[a]	Agent Name and Damage Type	Primary Tree Species Affected	Mapped Area (km^2)
11006	mountain pine beetle (*Dendroctonus ponderosae*); tree mortality	five-needle pines (*Pinus aristata, Pinus flexilis, Pinus strobiformis*), lodgepole pine (*Pinus contorta*), ponderosa pine (*Pinus ponderosa*)	10,614.3
12040	western spruce budworm (*Choristoneura freemani*); tree defoliation and occasional mortality	Douglas-fir (*Pseudotsuga menziesii*), subalpine fir (*Abies lasiocarpa*), white fir (*Abies concolor*), occasionally Engelmann spruce (*Picea engelmannii*)	9,437.0
80002	subalpine fir mortality[b] (*Dryocoetes confusus*, *Armillaria* spp., and others); tree mortality	subalpine fir (*Abies lasiocarpa*)	7,533.2
11009	spruce beetle (*Dendroctonus rufipennis*); tree mortality	blue spruce (*Picea pungens*), Engelmann spruce (*Picea engelmannii*)	6,071.1
80001	aspen defoliation (NA); tree defoliation and occasional mortality	aspen (*Populus tremuloides*)	4,333.4
11007	Douglas-fir beetle (*Dendroctonus pseudotsugae*); tree mortality	Douglas-fir (*Pseudotsuga menziesii*)	2,331.0
24032	sudden aspen decline (NA); tree dieback and mortality	aspen (*Populus tremuloides*)	1,557.5
11050	fir engraver (*Scolytus ventralis*); tree mortality	Douglas-fir (*Pseudotsuga menziesii*), subalpine fir (*Abies lasiocarpa*), white fir (*Abies concolor*); occasionally Engelmann spruce (*Picea engelmannii*)	1,420.5
11015	western balsam bark beetle (*Dryocoetes confusus*); tree mortality	subalpine fir (*Abies lasiocarpa*)	1,333.9
11019	pinyon ips (*Ips confusus*); tree mortality	two-needle pinyon pine (*Pinus edulis*)	1,248.8
25036	marssonina leaf blight (*Drepanopeziza punctiformis*); tree defoliation and occasional dieback and mortality	aspen (*Populus tremuloides*) and cottonwoods (*Populus angustifolia, Populus deltoides, Populus fremontii*)	1,178.5
11029	pine engraver (*Ips pini*); tree mortality	lodgepole pine (*Pinus contorta*), ponderosa pine (*Pinus ponderosa*)	689.7
80004	pinyon pine mortality (NA); tree mortality	two-needle pinyon pine (*Pinus edulis*)	666.3

12094	western tent caterpillar (*Malacosoma californicum*); tree defoliation and occasional mortality	aspen (*Populus tremuloides*), cottonwoods (*Populus angustifolia, Populus deltoides, Populus fremontii*), willows (*Salix* spp.)	525.3
80003	five-needle pine decline (NA); tree mortality	five-needle pines (*Pinus aristata, Pinus flexilis, Pinus strobiformis*)	338.9
11002	western pine beetle (*Dendroctonus brevicomis*); tree mortality	ponderosa pine (*Pinus ponderosa*)	314.5
12050	needle miner (*Coleotechnites* spp.); tree defoliation	various species, primarily Pinus	289.5
12180	tent caterpillars (*Malacosoma* spp.); tree defoliation and occasional mortality	various deciduous species (*Populus* spp., *Quercus gambelii*)	189.7
12123	Douglas-fir tussock moth (*Orgyia pseudotsugata*); tree defoliation and occasional mortality	blue spruce (*Picea pungens*), Douglas-fir (*Pseudotsuga menziesii*), subalpine fir (*Abies lasiocarpa*), white fir (*Abies concolor*); Engelmann spruce (*Picea engelmannii*)	185.2
25034	Lophodermella needle cast (*Lophodermella* spp.); tree defoliation	various pine species (*Pinus* spp.)	135.4

[a]Agent code corresponds to "DCA_CODE" field in Aerial Detection Survey Polygons 1997-2019.
[b]Subalpine fir mortality refers to subalpine fir decline (SFD), a mortality complex that includes western balsam bark beetle, *Armillaria* root rot, and other damage agents.

Table S2: Damage agent and host combinations used to restrict regional maps of bark beetle occurrence and severity in the Southern Rocky Mountains, USA. The study area was limited to subalpine forests on US Forest Service lands within 500 m of ADS polygons representing the following agent-host combinations.

Agent Name	**Tree Host Species**
mountain pine beetle (*Dendroctonus ponderosae*)	five-needle pines (*Pinus aristata, Pinus flexilis, Pinus strobiformis*), lodgepole pine (*Pinus contorta*)
spruce beetle (*Dendroctonus rufipennis*)	blue spruce (*Picea pungens*), Engelmann spruce (*Picea engelmannii*), spruce species (*Picea* spp.)
subalpine fir mortality (*Dryocoetes confusus*, *Armillaria* spp., and others)	subalpine fir/corkbark fir (*Abies lasiocarpa*)
Western balsam bark beetle (*Dryocoetes confusus*, *Armillaria* spp., and others)	subalpine fir/corkbark fir (*Abies lasiocarpa*)

Limiting the Study Area to Subalpine Forests Potentially Affected by Bark Beetles

We used several spatial datasets to restrict the study area to subalpine forests that may have experienced bark beetle-induced tree mortality, and to limit the influence of other forest disturbances (e.g., fire, timber harvests). First, we identified areas that could have been affected by bark beetle attack in the Southern Rocky Mountains (SRM), USA as all locations within 500 m of 1997-2019 ADS polygons with agent-tree host combinations described in Table S2 [1,2], a total area of 53,827 km^2. A 500-m buffer is a conservative threshold adopted in other studies that limits the omission of bark beetle-induced tree mortality due to locational errors in ADS data [3]. Next, we restricted these sites to locations that were classified as forest in the 1992 National Land Cover Dataset (NLCD) [4] and any vegetation class in 2016 NLCD [5], thereby removing developed lands, bare ground, and other unvegetated cover types, further limiting the study area to 39,573 km^2. To constrain our analyses to subalpine forests and account for incorrect agents or host codes in ADS data, we used species abundance maps [6] of lodgepole pine, Engelmann spruce, subalpine fir, and five-needle species, the dominant coniferous tree species in the subalpine zone. We summed values from individual species abundance maps and masked any locations in the study area with less than 1 $m^2\ ha^{-1}$ basal area combined across species. After masking to subalpine forests, the study area was 37,316 km^2. To limit the influence of fires and timber harvests that occurred 1996-2019, we excluded burned areas in the Monitoring Trends in Burn Severity (MTBS) [7] and geospatial multi-agency coordination (GeoMAC) [8] datasets, and removed recorded timber harvests, fuels treatments, sanitation treatments, and salvage logging in the national Forest Activity Tracking System (FACTS) dataset [9] and a regional fuels treatment database [10], reducing the study area to 33,275 km^2. Finally, we limited the study area to locations within US Forest Service boundaries, where descriptions of timber harvests were

most reliable. In total, the final study area spanned 25,946 km^2 with a 30-m grain size. So, while removing fire, timber harvest, and limiting the study area to USFS lands excludes some areas that were potentially affected by beetles, it retains 70% of the total area (i.e., subalpine forests within 500 m of ADS-mapped bark beetle-induced tree mortality) defined using other criteria, and better ensures that our analyses are recording bark beetle attack.'

Table S3: Data contributor, sample size (n = number of plots), plot design, and data collection protocol for the 239 field plots used to predict bark beetle presence and severity across the Southern Rocky Mountains, USA. Note that while field protocol and plot sizes differed slightly among contributors, we found no evidence for prediction bias based on contributor, initial forest density, or other potentially meaningful covariates.

Contributor(s)	**n**	**Plot Design and Collection Protocol**[a]
Andrus	106	Square field plots (20 x 20 m) were established in areas exceeding c. 20% mortality of *Picea engelmannii* and *Abies lasiocarpa* due to bark beetle attack c. 2000-2012. Field data characterized tree species, condition (i.e., live or dead), and diameter for all individuals exceeding 4 cm in diameter at breast height (DBH, 1.37 m above ground level). For dead stems, tree mortality agent was also recorded. Surveys were completed in 2016-2017.
Chapman	53	Belt transects (4 x 50 m) were established 2010-2012 in areas that were dominated by lodgepole pine (*Pinus contorta*) and had ADS-mapped tree mortality attributed to mountain pine beetle. Plots were stratified by stand composition in an effort to capture a range of stand types. Field data include records of tree species, condition (i.e., live or dead), and diameter for all individuals exceeding 4 cm DBH. For dead individuals, tree mortality agent was also recorded.
Gill	8	At each plot location, three parallel belt transects (each 2 x 50 m) were permanently established in the early 2000s in areas that were relatively undisturbed but adjacent to areas affected by historical bark beetle outbreaks or fires in 2002. These plots were resurveyed in 2014, and they were affected by bark beetles between measurements. Field plots characterized species, condition (i.e., live or dead), and diameter for all trees exceeding 4 cm in diameter at breast height (DBH, 1.37 m above ground level).
Hart	20	Square field plots (20 x 20 m) characterized tree species, condition (i.e., live or dead), and diameter for all stems exceeding 4 cm DBH. For dead individuals, tree mortality agent was also recorded. Surveys were completed 2010-2013.
Harvey	31	Circular field plots (36-m diameter) were established in areas with ADS-mapped tree mortality due to subalpine fir decline or western balsam bark beetle (*Dryocoetes confusus*). Plots characterized tree species, condition (i.e., live or dead), and diameter for all individuals taller than 1.37 m above ground level. For dead stems, tree mortality agent was also recorded. Surveys were completed in 2016.
Veblen	21	Rectangular field plots ranging 70-1,764 m^2 were permanently established 1982-1986 and 2016 (only three plots were established in 2016). Smaller "gap" plots were installed to characterize forest dynamics in relatively open areas, but the majority of field plots (i.e., "large plots" of c. 0.25 ha) were installed in relatively homogeneous, unlogged stands. Large plots were allowed to vary in size in an effort to survey at least 250 live trees. Field plots characterized tree species, condition (i.e., live or dead), and diameter for all exceeding 4 cm in diameter at breast height (DBH, 1.37 m above ground level) and were resurveyed at regular intervals for tree mortality. For dead stems, tree mortality agent was also recorded.

a: Visual interpretations of Landsat time series and high-resolution imagery were used at each field plot location to ensure that there was no visible mortality that occurred after the time of field surveys but prior to 2019, confirming that subsequent disturbances were not present in LandTrendr-derived predictors (i.e., total spectral decline 1997-2019) that would have been excluded in reference data.

Table S4: Summary of forest structure, tree mortality, and Random Forest model accuracy in field plots collected by different data contributors. Pre-outbreak basal area gives the density of all species c. late 1990s. Mortality gives the percentage of initial stand basal area that died c. 1990s-2010s, primarily due to bark beetle attack, though other minor causes were also included (e.g., drought, competition, sudden aspen decline). Presence gives the percentage of field plots in which bark beetle-induced tree mortality was correctly detected using the Random Forest model of bark beetle presence, based on 10-fold cross-validation. Note that 'Harvey' and 'Veblen' plots were detected at lower rates, primarily because they had lower amounts of tree mortality and a weaker spectral signal. Severity gives the root-mean-squared error (RMSE) of predicted and observed values of percent basal area mortality using Random Forest model of bark beetle severity, based on 10-fold cross-validation.

Contributor(s)	**Dominant Mortality Agent(s)**	**Pre-outbreak Basal Area (m^2 ha^{-1}); Mean (Min-Max)**	**Mortality (% BA); Mean (Min-Max)**	**Presence (%ACC)**	**Severity (RMSE)**
Andrus	SB, SFD	53.8 (15.9-104.2)	76.8 (26.2-98.4)	94.3	18.5
Chapman	MPB	63.8 (14.2-107.4)	72.7 (37.7-99.3)	86.8	14.3
Gill	MPB, SB, SFD	59.6 (36.9-80.9)	49.6 (4.5-91)	62.5	11.7
Hart	SB, SFD	58.9 (26.4-113.4)	66.6 (0.7-99.7)	70.0	22.9
Harvey	MPB, SB, SFD	74.3 (45.5-144.9)	27.2 (5.1-66.8)	19.4	18.1
Veblen	MPB, SB, SFD	72.5 (18.5-116.8)	13.1 (1.8-36.3)	33.3	11.3
All	MPB, SB, SFD	60.9 (14.2-144.9)	62.1 (0.7-99.7)	74.5[a]	17.3

MPB: Mountain pine beetle (*Dendroctonus ponderosae*)
SB: Spruce beetle (*Dendroctonus rufipennis*)
SFD: Subalpine fir decline (*Dryocoetes confusus*, *Armillaria* spp., and others)
[a]Percent classification accuracy for "all" plots does not include control points derived from aerial image interpretation and corresponds to specificity (i.e., correct detection given occurrence) of occurrence model. Overall accuracy when including control absence points was c. 80%.

Calculation of Winter NDVI

Many of the spectral bands and indices that we used to develop LTS were calculated from growing season imagery collected June 1-September 30. However, imagery from the winter season may help to isolate the signals of mortality and growth of evergreen conifers, distinct from herbaceous vegetation and deciduous tree species [11,12]. Therefore, we also developed annual composites of the Normalized Difference Vegetation Index (NDVI) from December $1_{\text{focal yr}}$ to April $1_{\text{focal yr +1}}$ for use with LandTrendr. Instead of the pre-processing routines typically used with imagery collected during the summer growing season [13], we used a separate set of pre-processing steps to calculate winter NDVI, following [12,14]. First, we restricted individual winter Landsat scenes to areas with "snow-on" conditions (based on a minimum value of 0.4 in the Normalized Difference Forest Snow Index [NDFSI]); [15]. After masking areas with the absence of snow cover, we developed yearly winter-season composites of NDVI using the 75^{th} percentile of NDVI values from all snow-on pixels in a given 30-m cell. The use of the 75^{th} percentile of NDVI, in combination with the snow mask developed from NDFSI, ensured that some canopy vegetation was visible but that partial snow cover obstructed much of the herbaceous vegetation below the forest canopy.

Spatial Segmentation of Annual Spectral Band and Index Values

LandTrendr is a pixel-based algorithm that partitions Landsat time series (LTS) into homogeneous periods of vegetation growth, stability, and decline [16]. As such, the spatial context surrounding each pixel is not explicitly incorporated into temporal segmentation using LandTrendr. However, when objects of interest exceed the area of an individual 30-m pixel (e.g., agricultural fields, forest stands), spatial segmentation and object-based image analysis may improve the identification of changes in surface cover using LTS [17–21]. Furthermore, spatial

segmentation can reduce high-frequency noise present in otherwise homogeneous areas, leading to more accurate and consistent end products (e.g., [22,23]). To incorporate spatial context in the development of LandTrendr products, we used Simple Non-Iterative Clustering (SNIC) [24] to develop smoothed annual maps of each band/index prior to temporal segmentation. For each spectral band and index (Table 1 in the main text), we applied SNIC at three spatial scales (5-, 10-, and 20-pixel seed spacing), where scale is defined as the spacing between 'seed locations' in a uniform grid used to initiate the development of an image object composed of adjacent, spectrally similar pixels. Following the identification of objects within each map using SNIC, individual pixel values (within a yearly composite) were reassigned to the mean value of all cells within the object. Thus, SNIC performed at a 20-cell seed spacing creates relatively large homogeneous objects within an annual map, while a 5-cell seed spacing retains greater fine-scale detail (Figure S1). Following spatial segmentation of annual maps, we developed four yearly time series of each spectral band and index (i.e., 16 spectral bands and indices for a total of 64 individual time series) within each 30-m voxel using (1) original, unsegmented maps (which is common practice when using LandTrendr), as well as the SNIC-smoothed annual maps at (2) 5-, (3) 10-, and (4) 20-pixel seed spacings. While LandTrendr temporal segmentation of yearly time series was still performed at the pixel-level, the smoothed annual maps used to develop the yearly time series in each 30-m voxel incorporate the spatial context of the surrounding area.

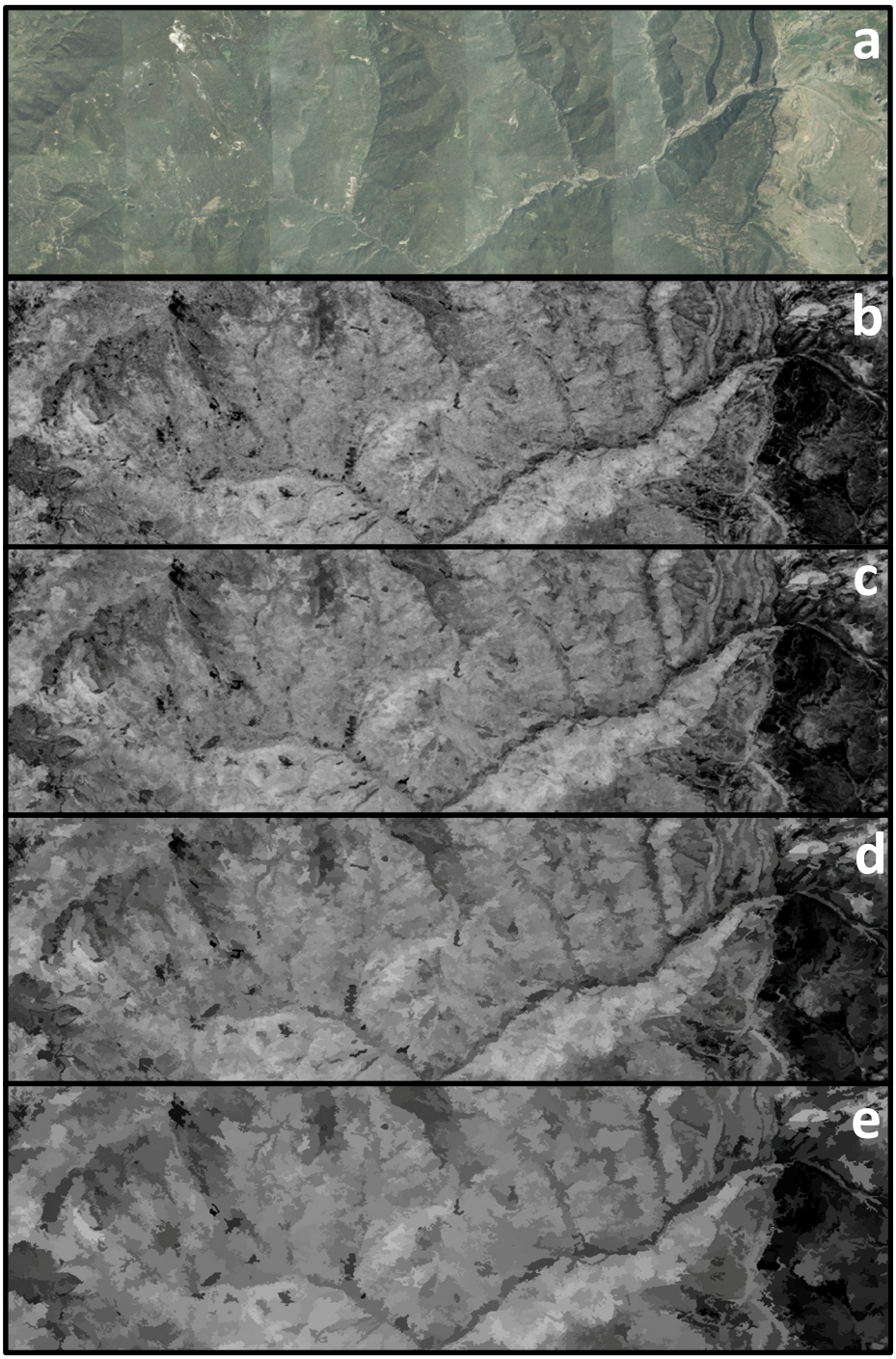


Figure S1: A comparison of (a) 2005 NAIP (National Agriculture Imagery Program) aerial imagery [25] and Landsat-derived maps of the Normalized Burn Ratio (NBR), with and without spatial segmentation. Annual maps of each spectral band and index were spatially segmented at multiple scales (5-, 10, and 20-pixel seed spacings) prior to temporal segmentation using LandTrendr. In the example scene, NBR for the 2005 growing season is shown (b) without spatial segmentation (i.e., raw NBR values), and with spatial segmentation at (c) 5-, (d) 10-, (e) 20-pixel seed spacings to illustrate the effects of segmentation scale.

Landsat ETM+ Data Gaps and Striping Patterns

Preliminary analyses suggested that including Landsat ETM+ images acquired after the scan-line corrector failure in 2003 led to characteristic striping patterns in regional maps of bark beetle occurrence and severity (Figure S2). Therefore, we compared maps with and without the inclusion of Landsat ETM+ images collected 2003-2019. The exclusion of Landsat ETM+ images from this period eliminated these striping artifacts while maintaining nearly identical patterns (Figure S2). Notably, this restriction also leads to a lack of scenes in the 2012 composite image for the study area, as ETM+ was the only Landsat sensor that was operational in 2012. However, the LandTrendr algorithm can effectively interpolate missing values in these cases by using the overall trend in the time series. Comparisons indicated that this data gap did not cause issues in final predictions, even when disturbance events (e.g., fire, harvest, bark beetle outbreak) occurred 2011-2013.

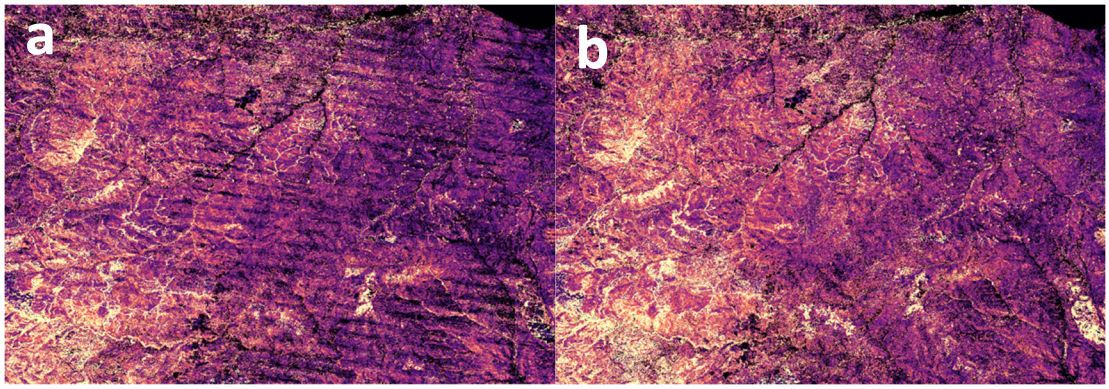


Figure S2: A comparison of Random Forest-derived maps of bark beetle severity in a portion of the study area, developed using Landsat time series products (a) with and (b) without the inclusion of Landsat ETM+ imagery acquired following the 2003 scan line corrector failure. Note that maps are nearly identical with the exception that striping (created by data gaps in individual scenes following the 2003 scan line corrector failure, leading to large within-scene variation in the seasonality of imagery) is removed by excluding Landsat ETM+ data 2003-2019 from annual image composites.

Identifying Slope Thresholds to Exclude Other Disturbances and Reduce Spectral Noise

While we used ancillary data to reduce the influence of fires and timber harvests on regional maps of bark beetle presence and severity, these databases are do not include some smaller fire events or unrecorded harvests. Thus, we also used a time series segment-level classification within LandTrendr to reduce the effect of unrecorded disturbance events unrelated to bark beetles. Spectral declines associated with bark beetle-induced tree mortality typically happen at a lower rate (i.e., spectral change yr^{-1}) than spectral declines due to wildfire or timber harvest [29]. Therefore, we excluded individual disturbance segments, identified through LandTrendr, that occurred with a greater rate of spectral change than was typical of bark beetle attack following [30]. For each spectral band/index and spatial segmentation combination, we calculated the rate of change in the largest LandTrendr-identified disturbance segment (hereafter "maximum slope") at each of 239 field plots with bark beetle-induced tree mortality c. 1990s-2010s. We compared the maximum slope values from these bark beetle plots with the maximum slope values in 107 field plots in subalpine forests that experienced wildfire 2012-2013 [31] and 97 points (located through the interpretation of imagery and LTS) in areas with timber harvest c. 1990s-2010s. These comparisons indicated that a maximum slope filter based on the common definition of statistical outliers (i.e., the 75^{th} percentile + 1.5 * the interquartile range of maximum slope values from beetle plots), applied during LandTrendr segmentation of each spectral band/index and spatial segmentation combination, removed many high-severity fire and timber harvest segments while still allowing the remaining time series to be utilized in a 30-m voxel (Fig S3; Table S5). In other words, locations with LandTrendr-identified segments exceeding the maximum slope filter were still retained in regional maps, but individual disturbance segments that exceeded the maximum slope filter were excluded.

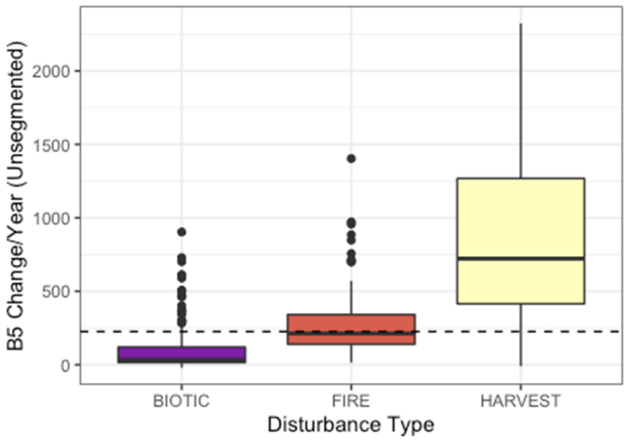


Figure S3: Example of maximum change slopes of largest disturbance identified by LandTrendr segmentation of TM-equivalent Band 5 across different disturbance types. As expected, biotic disturbances (i.e., bark beetle attack) showed a lower rate of spectral change yr^{-1} than did fire and harvest events. The dashed line gives the maximum slope threshold for bark beetle attack (257.7 in the case of B5 without spatial segmentation; Table S5), and disturbance segments with a rate of change exceeding the slope threshold were excluded from LandTrendr products to reduce noise and remove severe tree mortality that was unrelated to bark beetle attack (e.g., unidentified fires or harvest events). Note that the locations of known fires and timber harvests were also excluded from regional maps.

Table S5: Maximum slope thresholds, specific to each spectral band/index and spatial segmentation combination, used to filter disturbances that exceeded the typical rate of change (maximum spectral change yr^{-1}) for bark beetle attack. Thresholds were identified using the common definition of statistical outliers (75^{th} percentile + 1.5 * interquartile range) based on spectral change in the largest disturbance segment extracted from 239 field plots with bark beetle-induced tree mortality. Numbered subscripts correspond to the spatial scale of segmentation used with annual band/index maps (Figure S1) prior to temporal segmentation with LandTrendr (band/index values without subscripts were not spatially segmented).

Band/Index	Slope Threshold (change yr^{-1})	Accuracy (%)	BB Retained (%)	Fire/Harvest Removed (%)
B1	22.3	72	84.5	57.4
$B1_5$	44.3	64.6	85.4	40.2
$B1_{10}$	25.7	68.6	80.8	54.4
$B1_{20}$	27.3	65.9	79.9	49.5
B2	62.5	64.6	84.1	41.7
$B2_5$	100.8	62.3	85.4	35.3
$B2_{10}$	59.1	64.3	84.1	41.2
$B2_{20}$	51.8	61.4	82.4	36.8
B3	115.5	66.6	90.4	38.7
$B3_5$	121.4	64.3	88.7	35.8
$B3_{10}$	93.4	62.8	83.7	38.2
$B3_{20}$	95	58.2	81.6	30.9
B4	212.5	51.2	82	15.2
$B4_5$	223.8	51.2	81.6	15.7
$B4_{10}$	184.7	52.1	81.6	17.6
$B4_{20}$	198.2	49.4	80.3	13.2
B5	257.7	74.3	87	59.3
$B5_5$	258.7	74	90	55.4
$B5_{10}$	263.3	69.1	90	44.6
$B5_{20}$	156.8	72.9	86.2	57.4
B7	196.6	78.8	87.9	68.1
$B7_5$	213.3	78.8	88.3	67.6
$B7_{10}$	174.2	73.8	83.3	62.7
$B7_{20}$	171.5	74	84.1	62.3
EVI	148.8	60.3	82.8	33.8
EVI_5	77.3	62.1	81.2	39.7
EVI_{10}	180.2	57.8	83.3	27.9
EVI_{20}	170.2	54.6	82.8	21.6
NBR	159.8	71.3	92.1	47.1
NBR_5	151.9	70	91.6	44.6
NBR_{10}	159.9	66.8	89.5	40.2
NBR_{20}	122.4	66.4	84.5	45.1
NDMI	123.1	63.9	86.2	37.7
$NDMI_5$	149.9	62.8	88.3	32.8
$NDMI_{10}$	144.6	60.5	88.3	27.9
$NDMI_{20}$	161.4	59.8	90.8	23.5
NDVI	119.9	60	91.6	23
$NDVI_5$	118.3	60.3	91.2	24
$NDVI_{10}$	71.9	60.7	84.5	32.8

NDVI 20	136.2	55.5	93.7	10.8
RGI	75.8	58.2	88.7	22.5
RGI5	75.9	56.9	90.4	17.6
RGI 10	81.4	54.9	90.4	13.2
RGI 20	66.9	55.5	89.5	15.7
TCA	654.6	64.8	92.5	32.4
TCA5	472.4	63.9	84.1	40.2
TCA 10	712.2	62.5	90.4	29.9
TCA 20	424.6	59.8	82.8	32.8
TCB	152.1	76.3	83.3	68.1
TCB5	168.3	73.4	83.3	61.8
TCB 10	158	75.4	85.8	63.2
TCB 20	200	67.5	84.9	47.1
TCG	202.6	56.9	85.4	23.5
TCG5	239.6	54.2	85.8	17.2
TCG 10	138	56.9	80.8	28.9
TCG 20	168.4	57.3	84.9	25
TCW	400.3	74.5	94.1	51.5
TCW5	285.3	73.8	90.4	54.4
TCW 10	345.1	74.5	91.6	54.4
TCW 20	225.1	72.2	83.7	58.8
WinterNDVI	107.6	52.4	82	17.6
WinterNDVI5	99.9	52.4	82	17.6
WinterNDVI 10	82.5	51.9	78.2	21.1
WinterNDVI 20	195.3	49.9	86.2	7.4

Correcting Bias in Random Forest Predictions of Outbreak Severity

Ra

ndom Forest (RF) regression can reduce predicted values towards the mean of the response, biasing predictions at the tails of the distribution [32,33]. Preliminary analyses indicated that this was also a problem in the present study (Figure S4a), where low observed values of outbreak severity were overpredicted and high observed values were underpredicted (in 10-fold cross-validation) by fitted RF models. In the present study, accurate predictions of extreme values of outbreak severity were particularly important because low values may help to identify disturbance refugia and high values are indicative of near-total loss of the tree canopy. Thus, we corrected RF predictions of severity using the following equation, which is based on a common method of bias correction used with spatially-explicit climate records [34–36]:

$$\hat{y}_c = \left[\left(\frac{\hat{y}_{unc} - \mu_{rf}}{\sigma_{rf}}\right) \times \sigma_{obs}\right] + \mu_{obs} \quad (1)$$

where $\hat{y}_{unc}$ and $\hat{y}_{cor}$ are the predicted pixel values at a given field plot, before and after bias correction, respectively. μ_{rf} and μ_{obs} are the means and σ_{rf} and σ_{obs} are the standard deviations of RF-predicted and observed pixel values across all field plots in 10-fold cross-validation (where ten separate RF models were constructed, each trained using 90% of the data and tested on the remaining 10%). When bias correction led to predictions above 100 or less than 0, these values were truncated at 100 and 0, respectively. Here, μ_{rf} was 61.66, μ_{obs} was 62.08, σ_{rf} was 22.06, and σ_{obs} was 29.25. These values indicate that the RF model was accurately predicting the mean and underpredicting the variance of the distribution. The results of this bias correction are presented in Figure S4b and S4c, where corrected values (i.e., 'predBC' in Figure S4c) better match the statistical distribution of the observed data than do uncorrected values. While bias correction

slightly increased the RMSE of RF model predictions (c. 0.6 increase in RMSE), it led to more realistic predictions at the ends of the distribution. The correlation between observed and predicted values was unaffected because our bias correction scales the predicted values, thus maintaining the relative relationships within the predicted and observed values.

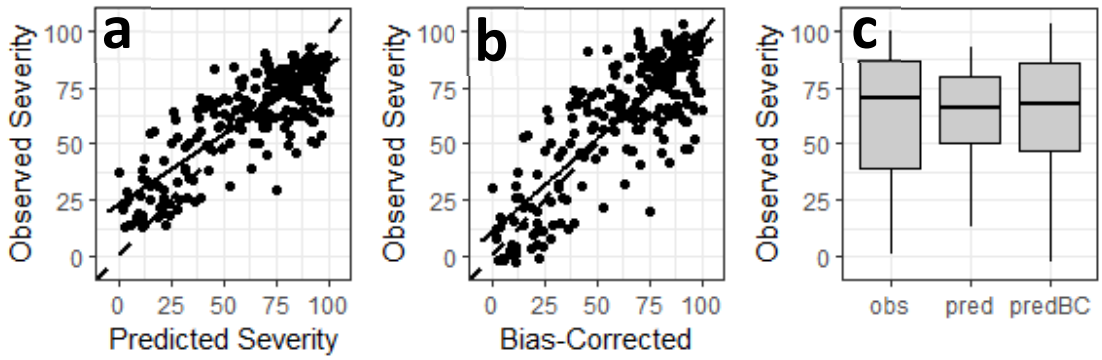


Figure S4: A comparison of predicted and observed values of bark beetle outbreak severity (i.e., cumulative percent basal area loss 1997-2019) at the 239 field plots in the Southern Rocky Mountains, USA. Predicted values were derived from a Random Forest model using predictors derived from Landsat time series. Panel (a) gives the relationship between RF-predicted and observed values in field plots, based on 10-fold cross-validation. Panel (b) shows this same relationship after bias correction. In (a, b), the dashed line represents a 1:1 relationship, while the solid line shows a linear fit to the observed relationship. Panel (c) shows the statistical distributions of observed (obs) values in field data, RF-predicted values (pred), and RF-predicted values following bias correction (predBC).

References

1. US Forest Service Insect and Disease Detection Surveys Available online: https://www.fs.fed.us/foresthealth/applied-sciences/mapping-reporting/detection-surveys.shtml#idsdownloads (accessed on Mar 15, 2020).
2. Johnson, E.W.; Wittwer, D. Aerial detection surveys in the United States. *Aust. For.* **2008**, *71*, 212–215, doi:10.1080/00049158.2008.10675037.
3. Coleman, T.W.; Graves, A.D.; Heath, Z.; Flowers, R.W.; Hanavan, R.P.; Cluck, D.R.;

Ryerson, D. Accuracy of aerial detection surveys for mapping insect and disease disturbances in the United States. *For. Ecol. Manage.* **2018**, *430*, 321–336, doi:10.1016/j.foreco.2018.08.020.
4. Wickham, J.D.; Stehman, S. V.; Smith, J.H.; Yang, L. Thematic accuracy of the 1992 National Land-Cover Data for the western United States. *Remote Sens. Environ.* **2004**, *91*, 452–468, doi:10.1016/j.rse.2004.04.002.
5. Homer, C.; Dewitz, J.; Jin, S.; Xian, G.; Costello, C.; Danielson, P.; Gass, L.; Funk, M.; Wickham, J.; Stehman, S.; et al. Conterminous United States land cover change patterns 2001–2016 from the 2016 National Land Cover Database. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 184–199, doi:10.1016/j.isprsjprs.2020.02.019.
6. Wilson, B.T.; Lister, A.J.; Riemann, R.I.; Griffith, D.M. *Live tree species basal area of the contiguous United States (2000-2009)*; St. Paul, MN, USA, 2013;
7. Eidenshink, J.; Schwind, B.; Brewer, K.; Zhu, Z.; Quayle, B.; Howard, S.; Falls, S.; Falls, S. A Project for Monitoring Trends in Burn Severity. *Fire Ecol.* **2007**, *3*, 3–21.
8. Geospatial Multi-agency Coordinating Group (GeoMAC) Available online: https://rmgsc.cr.usgs.gov/outgoing/GeoMAC/ (accessed on Feb 25, 2020).
9. USDA Forest Service Geodata - Downloadable National Datasets Available online: https://data.fs.usda.gov/geodata/edw/datasets.php (accessed on May 20, 2020).
10. Caggiano, M.D. *Front Range Round Table 2016 Interagency Fuel Treatment Database*; Fort Collins, CO, 2017;
11. Baker, E.H.; Painter, T.H.; Schneider, D.; Meddens, A.J.H.; Hicke, J.A.; Molotch, N.P. Quantifying insect-related forest mortality with the remote sensing of snow. *Remote Sens. Environ.* **2017**, *188*, 26–36, doi:10.1016/j.rse.2016.11.001.
12. Vanderhoof, M.K.; Hawbaker, T.J.; Ku, A.; Merriam, K.; Berryman, E.; Cattau, M. Tracking rates of post-fire conifer regeneration distinct from deciduous vegetation recovery across the western USA. *Ecol. Appl.* **2020**, eap.2237, doi:10.1002/eap.2237.
13. Kennedy, R.E.; Yang, Z.; Gorelick, N.; Braaten, J.; Cavalcante, L.; Cohen, W.B.; Healey, S. Implementation of the LandTrendr Algorithm on Google Earth Engine. *Remote Sens.* **2018**, *10*, 691.
14. Vanderhoof, M.K.; Hawbaker, T.J. It matters when you measure it: Using snow-cover Normalised Difference Vegetation Index (NDVI) to isolate post-fire conifer regeneration. *Int. J. Wildl. Fire* **2018**, *27*, 815–830, doi:10.1071/WF18075.
15. Wang, X.Y.; Wang, J.; Jiang, Z.Y.; Li, H.Y.; Hao, X.H. An effective method for snow-cover mapping of dense coniferous forests in the upper Heihe River Basin using Landsat Operational Land Imager data. *Remote Sens.* **2015**, *7*, 17246–17257, doi:10.3390/rs71215882.
16. Kennedy, R.E.; Yang, Z.; Cohen, W.B. Detecting trends in forest disturbance and recovery using yearly Landsat time series: 1. LandTrendr - Temporal segmentation algorithms. *Remote Sens. Environ.* **2010**, *114*, 2897–2910, doi:10.1016/j.rse.2010.07.008.
17. Yin, H.; Prishchepov, A. V.; Kuemmerle, T.; Bleyhl, B.; Buchner, J.; Radeloff, V.C. Mapping agricultural land abandonment from spatial and temporal segmentation of Landsat time series. *Remote Sens. Environ.* **2018**, *210*, 12–24, doi:10.1016/j.rse.2018.02.050.
18. Gómez, C.; White, J.C.; Wulder, M.A. Characterizing the state and processes of change in a dynamic forest environment using hierarchical spatio-temporal segmentation. *Remote Sens. Environ.* **2011**, *115*, 1665–1679, doi:10.1016/j.rse.2011.02.025.

19. Hughes, M.J.; Douglas Kaylor, S.; Hayes, D.J. Patch-based forest change detection from Landsat time series. *Forests* **2017**, *8*, 1–22, doi:10.3390/f8050166.
20. Kennedy, R.E.; Yang, Z.; Braaten, J.; Copass, C.; Antonova, N.; Jordan, C.; Nelson, P. Attribution of disturbance change agent from Landsat time-series in support of habitat monitoring in the Puget Sound region, USA. *Remote Sens. Environ.* **2015**, *166*, 271–285, doi:10.1016/j.rse.2015.05.005.
21. Hermosilla, T.; Wulder, M.A.; White, J.C.; Coops, N.C.; Hobart, G.W. Regional detection, characterization, and attribution of annual forest change from 1984 to 2012 using Landsat-derived time-series metrics. *Remote Sens. Environ.* **2015**, *170*, 121–132, doi:10.1016/j.rse.2015.09.004.
22. Charoenjit, K.; Zuddas, P.; Allemand, P.; Pattanakiat, S.; Pachana, K. Estimation of biomass and carbon stock in Para rubber plantations using object-based classification from Thaichote satellite data in Eastern Thailand. *J. Appl. Remote Sens.* **2015**, *9*, 096072, doi:10.1117/1.jrs.9.096072.
23. Rodman, K.C.; Veblen, T.T.; Saraceni, S.; Chapman, T.B. Wildfire Activity and Land Use Drove 20th Century Changes in Forest Cover in the Colorado Front Range. *Ecosphere* **2019**, *10*, e02594, doi:10.1002/ecs2.2594.
24. Achanta, R.; Süsstrunk, S. Superpixels and Polygons using Simple Non-Iterative Clustering. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Institute of Electrical and Electronics Engineers: Honolulu, HI, 2017; pp. 4895–4904.
25. National Agriculture Imagery Progam, United States Forest Service Available online: http://www.fsa.usda.gov/programs-and-services/aerial-photography/imagery-programs/naip-imagery/ (accessed on Feb 8, 2020).
26. Bright, B.C.; Hudak, A.T.; Kennedy, R.E.; Braaten, J.D.; Henareh Khalyani, A. Examining post-fire vegetation recovery with Landsat time series analysis in three western North American forest types. *Fire Ecol.* **2019**, *15*, doi:10.1186/s42408-018-0021-9.
27. Senf, C.; Seidl, R. Mapping the forest disturbance regimes of Europe. *Nat. Sustain.* **2021**, *4*, 63–70, doi:10.1038/s41893-020-00609-y.
28. Genuer, R.; Poggi, J.-M.; Tuleau-Malot, C. Package "VSURF." *Pattern Recognit. Lett.* **2015**, *31*, 2225–2236.
29. Cohen, W.B.; Yang, Z.; Stehman, S. V.; Schroeder, T.A.; Bell, D.M.; Masek, J.G.; Huang, C.; Meigs, G.W. Forest disturbance across the conterminous United States from 1985-2012: The emerging dominance of forest decline. *For. Ecol. Manage.* **2016**, *360*, 242–252, doi:10.1016/j.foreco.2015.10.042.
30. Meigs, G.W.; Kennedy, R.E.; Gray, A.N.; Gregory, M.J. Spatiotemporal dynamics of recent mountain pine beetle and western spruce budworm outbreaks across the Pacific Northwest Region, USA. *For. Ecol. Manage.* **2015**, *339*, 71–86, doi:10.1016/j.foreco.2014.11.030.
31. Andrus, R.A.; Veblen, T.T.; Harvey, B.J.; Hart, S.J. Fire Severity Unaffected by Spruce Beetle Outbreak in Spruce-Fir Forests in Southwestern Colorado. *Ecol. Appl.* **2016**, *26*, 700–711, doi:10.1890/15-1121/suppinfo.
32. Zhang, G.; Lu, Y. Bias-corrected random forests in regression. *J. Appl. Stat.* **2012**, *39*, 151–160, doi:10.1080/02664763.2011.578621.
33. Parks, S.A.; Holsinger, L.M.; Koontz, M.J.; Collins, L.; Whitman, E.; Parisien, M.A.; Loehman, R.A.; Barnes, J.L.; Bourdon, J.F.; Boucher, J.; et al. Giving ecological meaning

to satellite-derived fire severity metrics across North American forests. *Remote Sens.* **2019**, *11*, 1–19, doi:10.3390/rs11141735.
34. Rodman, K.C.; Veblen, T.T.; Battaglia, M.A.; Chambers, M.E.; Fornwalt, P.J.; Holden, Z.A.; Kolb, T.E.; Ouzts, J.R.; Rother, M.T. A Changing Climate is Snuffing Out Post-Fire Recovery in Montane Forests. *Glob. Ecol. Biogeogr.* **2020**, *29*, 2039–2051, doi:10.1111/GEB.13174.
35. Flint, L.E.; Flint, A.L. Downscaling Future Climate Scenarios to Fine Scales for Hydrologic and Ecological Modeling and Analysis. *Ecol. Process.* **2012**, *1*, 1–15.
36. Bouwer, L.; Aerts, J.; Van de Coterlet, G.; Van de Giesen, N.; Gieske, A.; Mannaerts, C. Evaluating Downscaling Methods for Preparing Global Circulation Model (GCM) Data for Hydrological Impact Modeling. In *Climate Change in Contrasting Biver basins: Adaptation Strategies for Water, Food and Environment*; CAB International Publishing: London, 2004; pp. 25–47 ISBN 0851998356.

Knitted R Markdown Document Showing Fitting and Evaluation of Final Random Forest Models

Predicting Tree Mortality Due to Bark Beetle Activity in the Southern Rocky Mountains Using Field Data, Landsat Time Series, and A Stacked Ensemble Approach

Bark Beetle Outbreak Presence and Severity in the Southern Rocky Mountains

This markdown document presents analyses used to predict the presence and severity of bark beetle attack in the Southern Rocky Mountains (Colorado, southern Wyoming, and northern New Mexico), USA. The code below takes products from the LandTrendr temporal segmentation algorithm (Kennedy et al. 2010; Remote Sensing of Environment), calculated in Google Earth Engine using 16 different Landsat spectral bands and indices sensitive to vegetation, as well as different pre- and post-processing techniques, and compares these remotely-sensed products with field data to develop predictions of (1) presence/absence (occurrence) of disturbance and (2) percent basal area mortality (severity) across the entire region.

Prior to these analyses, we extracted the total disturbance magnitude (i.e., cumulative magnitude of spectral change from all disturbances 1997-2019 in each c. 30-m Landsat cell) across 64 different layers (i.e., LandTrendr products developed from 16 different bands and indices, each without spatial segmentation and with three different scales of spatial segmentation) in LandTrendr at each of 239 field plots and 239 “control points” that had no evidence of disturbance. Spatial segmentation (simple non-iterative clustering; Achantra and Süsstrunk 2017; computer vision and pattern recognition) at 5-, 10-, and 20-cell seed spacing used to smooth annual Landsat image composites (reducing noise in local neighborhoods) and slope thresholding was used to remove individual disturbance segments that were more severe and shorter in duration than we would expect of bark beetle outbreaks (similar to the approach used in Meigs et al. 2015; Forest Ecology and Management). We used the VSURF package in R to select the most parsimonious subset of LandTrendr-based predictors to predict occurrence and severity and created separate .csv files that included only the selected predictors, the response variable, and a few additional columns that were not included in the main analyses but have the potential to influence results (e.g., site ID, data contributor, pre-disturbance BA, field plot size, defoliation). These are imported near the top of this document.

After performing the analyses below, we used the fitted Random Forest models (in other scripts) to predict presence/absence and severity of disturbance throughout the Southern Rocky Mountains Ecoregion. Examples of these maps are shown in the main text and accompanying markdown document and also included in the data archive file via data dryad.

Links below go to different sections of the markdown document.

1) *Visualizing Field Data*

2) *Occurrence Model Fitting*

3) *Occurrence Model Summary*

4) *Severity Model Fitting*

5) *Severity Model Summary*

So, first bring in relevant packages

The following packages are used for data cleaning, model tuning, model fitting, and making maps/figures.

```
## Bring in Necessary Packages
package.list <- c("ranger", "caret", "ggplot2", "pdp", "here", "RColorBrewer"
,
                  "sjPlot", "cowplot", "sf", "leaflet", "doParallel", "paralle
l",
                  "tidyverse", "corrplot", "viridisLite", "rfUtilities")

## Installing them if they aren't already on the computer
new.packages <- package.list[!(package.list %in% installed.packages()[,"Packa
ge"])]
if(length(new.packages)) install.packages(new.packages)

## And loading them
for(i in package.list){library(i, character.only = T)}
```

Bring in field data

The code below brings in the formatted .csv files that include field data and extractions for each of the selected predictors derived from LandTrendr.

```
## Field data with percent mortality as well as aerial image interpretation
  # points without evidence of disturbance. Used in RF classification of dist
urbance
  # presence/absence
typeData <- read.csv(here("Data", "AnalysisReady", "typePoints_final.csv"))
  # Reformatting response to 1/0 so that RF can predict numeric value to rast
er maps
typeData$DisturbanceType <- as.numeric(typeData$DisturbanceType == "BIOTIC")

## Field data with percent mortality and extracted values of LandTrendr produ
ct.
  # Used in RF regression of disturbance severity
fieldData <- read.csv(here("Data", "AnalysisReady", "severityPoints_final.csv
"))
```

Set global parameters and create functions

The following code sets some global parameters (i.e., number of folds in cross-validation and colors used in plotting), and saves functions for later use.

```
## Setting number of folds for cross-validation in hyperparameter tuning and
  # accuracy assessment
nfolds <- 10

## Setting colors for plotting
bbColor <- "#7E03A8FF"
noDisturbColor <- "#FB8861FF"

## Function to set seeds in parallel processing for caret tuning
setSeeds <- function(method = "cv", numbers = 1, repeats = 1, tunes = NULL, s
eed = 71) {
  #B is the number of resamples and integer vector of M (numbers + tune lengt
h if any)
  B <- if (method == "cv") numbers
  else if(method == "repeatedcv") numbers * repeats
  else NULL

  if(is.null(length)) {
    seeds <- NULL
  } else {
    set.seed(seed = seed)
    seeds <- vector(mode = "list", length = B)
    seeds <- lapply(seeds, function(x) sample.int(n = 1000000, size = numbers
+ ifelse(is.null(tunes), 0, tunes)))
    seeds[[length(seeds) + 1]] <- sample.int(n = 1000000, size = 1)
  }
  # return seeds
  seeds
}
```

Visualizing locations and characteristics of field plots

The map below shows the locations of field plots with colors scaled from low (yellow) to high (purple) mortality during outbreaks c. 1997-2019. The map is interactive and allows zooming/panning. If you click on a specific point, it will show the data contributor(s), the percentage of tree basal area that died in the outbreak, and the pre-disturbance live basal area of the surveyed stand.

```
## First, create sf object of plots for mapping
  # Filter to important columns for map
sfPlots <- fieldData %>%
  select(SiteT:percentMortality)
  # Create SF object
sfPlots <- st_as_sf(x = sfPlots, coords = c("longitude", "latitude"),
```

```
                        crs = "+proj=longlat +datum=WGS84")

## Create labels for map
mapLab <- paste0("<strong>Data Contributor: </strong>", sfPlots$contributor,
"<br>",
                 "<strong>Basal Area Mortality (%): </strong>",
                 round(sfPlots$percentMortality, 1), "<br>",
                 "<strong>Pre-Outbreak Basal Area (sq m): </strong>",
                 round(sfPlots$preBA_HA, 1))

## Color palette for points
pal_fun <- colorBin(viridis(10, direction = -1), NULL, bins = 10)

## Making map
leaflet(sfPlots) %>%
  addTiles() %>%
  addCircleMarkers(
    color = ~pal_fun(percentMortality),
    popup = mapLab,
    stroke = FALSE, fillOpacity = 0.5
  )

## PhantomJS not found. You can install it with webshot::install_phantomjs().
If it is installed, please make sure the phantomjs executable can be found vi
a the PATH variable.
```

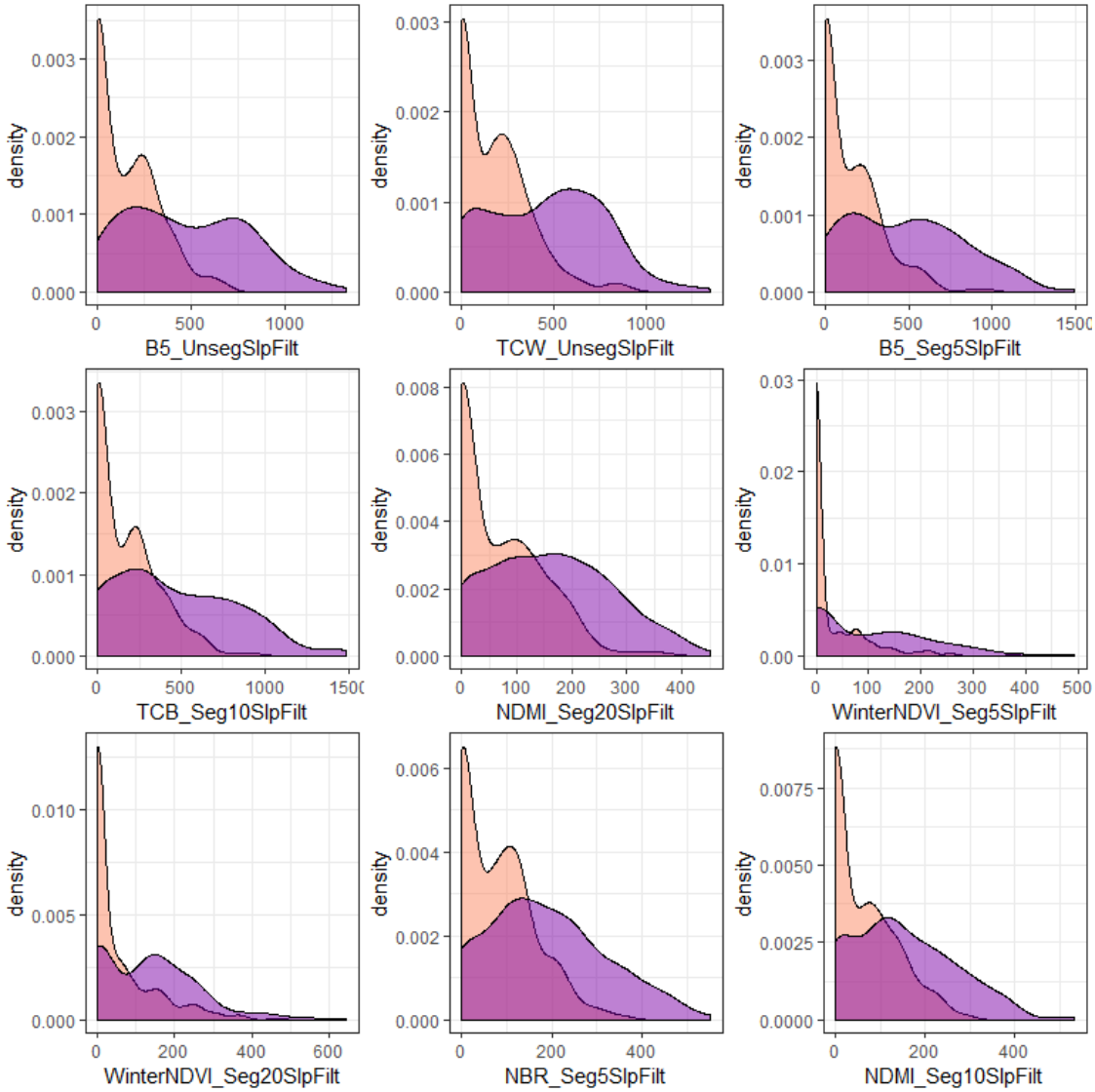
The occurrence model

Plotting distributions of the different variables in the presence/absence model

The plots below show the sampled distributions of each of the VSURF-selected predictors used to predict occurrence of bark beetle attack. Purple density plots are values of field plots with bark beetle-induced tree mortality, and orange density plots are control points that had no evidence of disturbance. Though bands and spectral indices have different responses to vegetation change (e.g., visible bands often increase in brightness while near-infrared bands decline), all of the predictors below are oriented such that higher values are indicative of greater decline in vegetation. Note that although VSURF selected six additional predictors for this model, they had evidence of overfitting because (1) disturbance occurrence was associated with little spectral change (the opposite of what we would expect), or (2) weird patterns in partial dependence plots demonstrated that a small number of outliers were causing local maxima/minima. The OOB classification error of the model sightly decreased with the removal of these six predictors.

The names of each variable follow the following syntax: (band/spectral index)_(if spatial segmentation was used “Seg” or not “Unseg”)(If spatial segmentation was performed, at what spatial scale? Higher numbers mean a coarser scale and larger objects)(“SlpFilt”)

```r
## Getting indices of columns
start <- which(colnames(typeData) == "DisturbanceType")
end <- ncol(typeData)

## Making plots of predictors based on presence/absence
densPlots <- lapply(colnames(typeData)[(start+1):end],
                    FUN = function(x, data = typeData){

  ## Plotting
  p <- ggplot(data, aes(x = data[,colnames(data) == x],
                        fill = as.factor(DisturbanceType))) +
    geom_density(color = "black", alpha = 0.5) +  xlab(as.character(x)) +
    ylab("density") + theme_bw() + theme(legend.position = "none") +
    scale_fill_manual(values = c(noDisturbColor, bbColor))

  ## Returning plot
  return(p)
})

## Putting list of plots together
sjPlot::plot_grid(densPlots, margin = c(0.1,0.1,0.1,0.1))
```

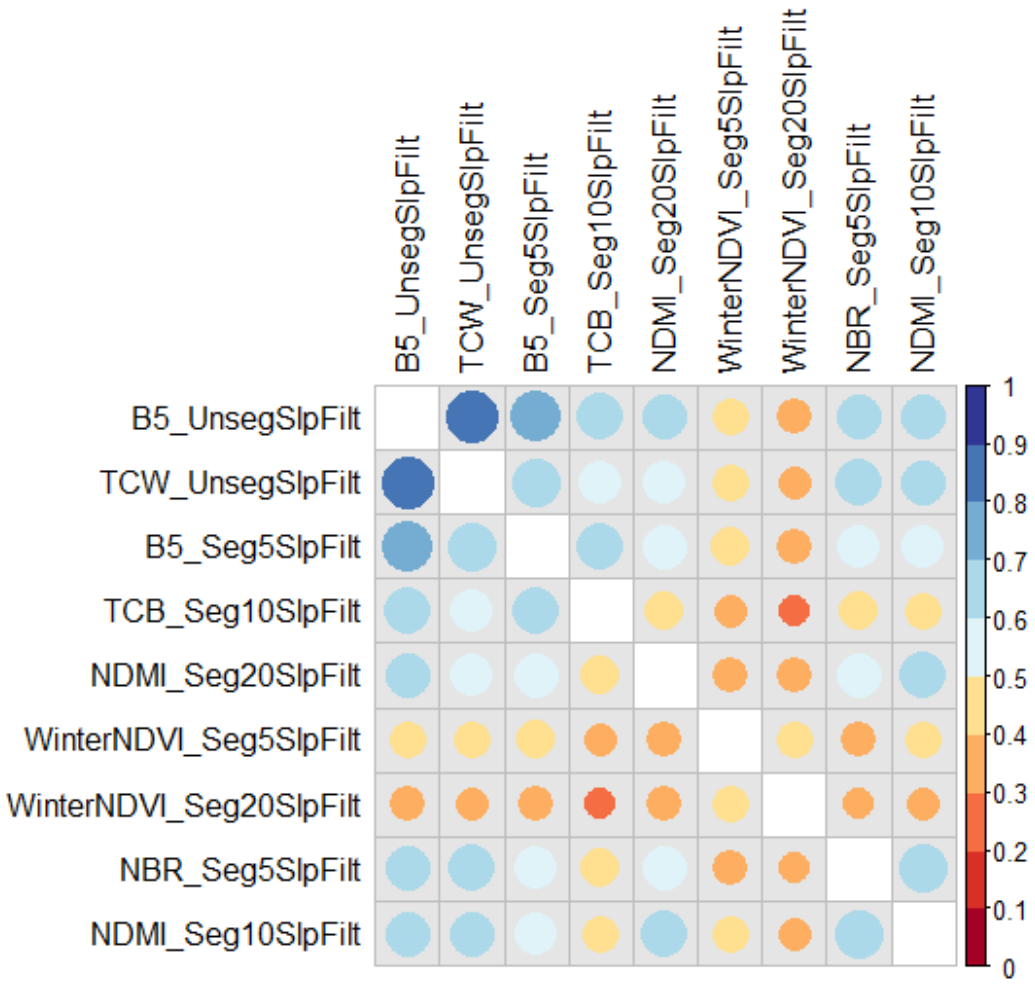


Correlation plot of each predictor of disturbance occurrence

The plot below shows the correlations among all of the predictors from LandTrendr. Because the response is binary (presence/absence), we exclude it from the correlation matrix, although differences in each predictor for disturbed and undisturbed sites are shown in the density plots above. The correlation among predictors isn't really a concern with the Random Forest model we are using for prediction, but it is interesting to see the similarities and differences among the different metrics. Note that all correlations are positive, with lower correlations in red, and higher correlations as blue.

```
## Creating pairwise correlation matrix
corrMat <- cor(typeData[c(start+1):end])

## Color ramp - hacky way to get the function below to work is to repeat twic
e
pal <- brewer.pal(10, name = "RdYlBu"); pal <- c(pal, pal)
```

```
## Making correlation plot
corrplot(corrMat, method = "circle", diag = F, cl.lim = c(0, 1),
         tl.col = "black", col = pal, bg = "grey90")
```



Tuning and fitting the Random Forest model of occurrence

The code chunk below tests to see if any additional variables can be excluded from the final model using recursive feature elimination, tunes hyperparameters (i.e., number of variables to split at each node and minimum node size), and fits the final model using the optimal parameter set.

```
## Getting subset of variables with predictors and response
modData <- typeData[,c(start:end)]

  # Convert response to factor for RF classification
```

```r
modData$DisturbanceType <- as.factor(modData$DisturbanceType)

## Testing multicollinearity of predictors
multi.collinear(modData[,-1])

## NULL

## Creating CV folds for tuning and fitting
set.seed(589)
folds <- createFolds(modData$DisturbanceType, k = nfolds, list = TRUE)

## Setting parameters for recursive feature elimination (i.e., backward selec
tion)
  # First establishing seeds for multi-core processing
seeds <- setSeeds(numbers = nfolds, tunes = ncol(modData)-1, seed = 341)
  # And setting parameters
controls <- rfeControl(method = "cv", number = nfolds,
                       index = folds, seeds = seeds, functions = rfFuncs)
  # Initiating parallel processing and setting seed
cl <- makeCluster(detectCores()-1)
registerDoParallel(cl)

## Running recursive feature elimination
set.seed(212)
caretRFE <- rfe(y = modData$DisturbanceType,
                x = within(modData, rm(DisturbanceType)),
                rfeControl = controls,
                sizes = ncol(modData)-1)
stopCluster(cl)
#caretRFE ## Including all terms is the best option here

# Set hyperparameters for tuning
controls <- trainControl(method = "cv", number = nfolds,
                         index = folds,
                         savePredictions = "all")
tgrid <- expand.grid(
  mtry = 2:5,
  splitrule = c("gini"),
  min.node.size = c(1, 5, 10, 25, 50, 100)
)

# Running model on tuning grid
set.seed(312)
cl <- makeCluster(detectCores()-1)
registerDoParallel(cl)
model_caret <- train(y = modData$DisturbanceType,
                     x = within(modData, rm(DisturbanceType)),
                     method = "ranger",
                     trControl = controls,
                     tuneGrid = tgrid,
```

```
                      importance = "permutation", num.trees = 1000)
stopCluster(cl)

## Getting optimal hyperparameters from cross-validation above
optMtry <- as.numeric(model_caret$bestTune[1,1])
optSplit <- as.character(model_caret$bestTune[1,2])
optMinNode <- as.numeric(model_caret$bestTune[1,3])

## Getting other variables to track in crossvalidation
mortAgent <- typeData$domDisturb; baPre <- typeData$preBA_HA
pctMort <- typeData$percentMortality; sbwTrees <- typeData$sbw_Trees
contrib <- typeData$contributor

## Getting seeds for crossvalidation
  # Creating separate folds for accuracy assessment
set.seed(192)
folds <- createFolds(modData$DisturbanceType, k = nfolds, list = TRUE)
set.seed(520)
cvSeeds <- sample(1:1000, nfolds)

## Performing 10-fold cross-validation of model
predVals <- c(); obsVals <- c(); domDist <- c()
prevBA <- c(); bbSev <- c(); sbwSev <- c(); contr <- c()
for(i in 1:nfolds){
  ## Subsetting data
  testData <- modData[folds[[i]],] ## Subsetting to test set
  trainData <- modData[-folds[[i]],] ## Subsetting to training set

  ## Getting other tracked values - contributor, mort agent, and previous BA
  domDist <- c(domDist, as.character(mortAgent[folds[[i]]]))
  prevBA <- c(prevBA, baPre[folds[[i]]])
  bbSev <- c(bbSev, pctMort[folds[[i]]])
  sbwSev <- c(sbwSev,  sbwTrees[folds[[i]]])
  contr <- c(contr, as.character(contrib[folds[[i]]]))

  ## Fitting model to training data and predicting to test set
  mod <- ranger(formula = DisturbanceType~., data = trainData,
                num.trees = 1000, min.node.size = optMinNode,
                mtry = optMtry, splitrule = optSplit, seed = cvSeeds[i])

  ## And getting those values
  obsVals <- c(obsVals, as.character(testData$DisturbanceType))
  predVals <- c(predVals, as.character(predict(mod, data = testData)$predicti
ons))
}

## Developing final RF model of disturbance severity with selected parameters
  # and full dataset
(presMod <- ranger(formula = DisturbanceType~., data = modData,
                   importance = "permutation", num.trees = 1000,
```

```
                  min.node.size = optMinNode, splitrule = optSplit,
                mtry = optMtry, seed = 727))

## Ranger result
##
## Call:
##  ranger(formula = DisturbanceType ~ ., data = modData, importance = "permu
tation",      num.trees = 1000, min.node.size = optMinNode, splitrule = optSp
lit,      mtry = optMtry, seed = 727)
##
## Type:                             Classification
## Number of trees:                  1000
## Sample size:                      478
## Number of independent variables:  9
## Mtry:                             2
## Target node size:                 25
## Variable importance mode:         permutation
## Splitrule:                        gini
## OOB prediction error:             20.08 %

#saveRDS(presMod, here("AnalysisOutputs", "BB_PresenceMod.rds"))
```

Summarizing results of the model of disturbance presence/absence

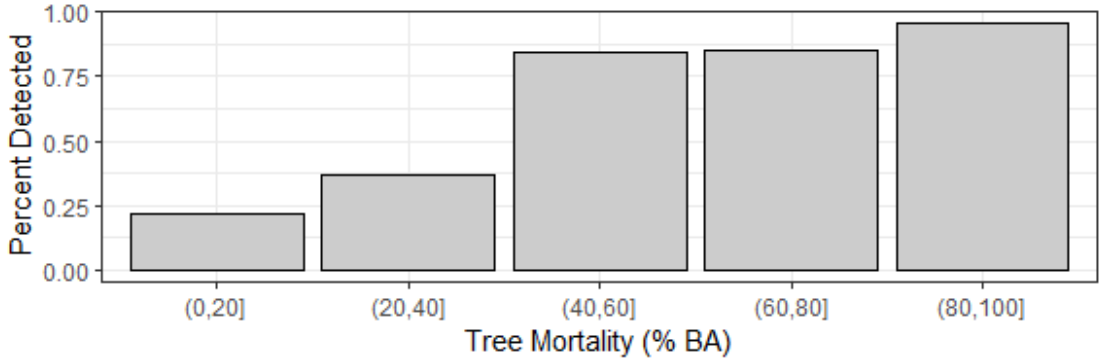
Overall, the model of occurrence model is performing OK (c. 80% accurate in OOB predictions). It is also worth noting that this is restricted to biotic disturbances that have a weaker spectral signal than fire/harvest. If we further restrict predictions using ADS data or other ancillary data sources, I would expect the accuracy to go up. The code below prints the classification accuracy of the model in 10-fold cross-validation and makes diagnostic plots that describe the model predictions by contributor, initial stand density, primary mortality agent, by percent tree mortality, and across a range of defoliation by spruce budworm.

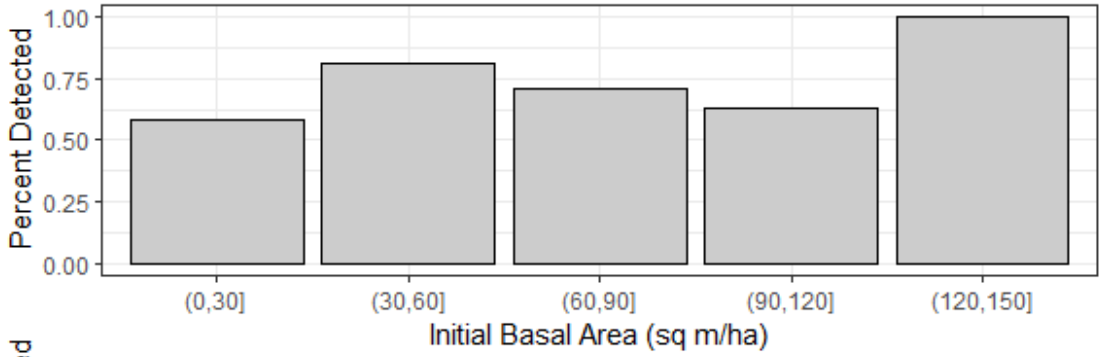
```
## Summarizing classification accuracy in final model
confusionMatrix(data = as.factor(predVals), reference = as.factor(obsVals))

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 206  61
##          1  33 178
##
##                Accuracy : 0.8033
##                  95% CI : (0.7648, 0.8381)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6067
##
##  Mcnemar's Test P-Value : 0.005355
```

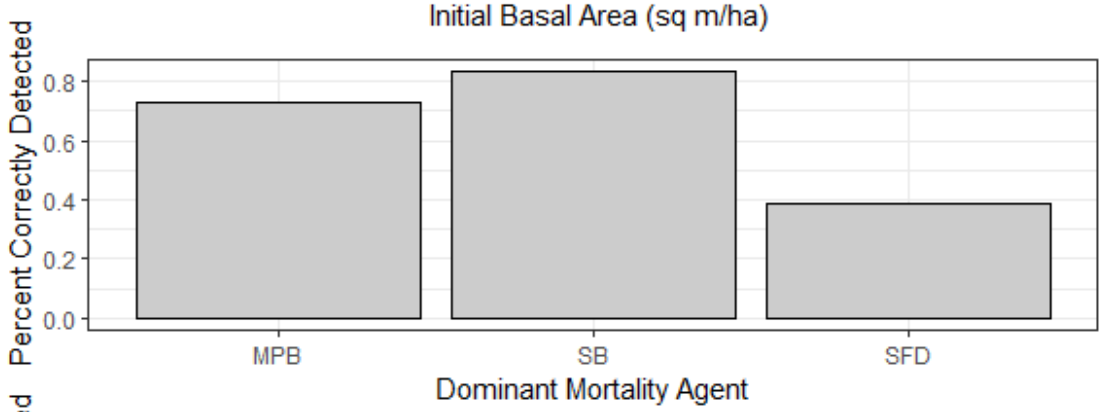
```
##
##             Sensitivity : 0.8619
##             Specificity : 0.7448
##          Pos Pred Value : 0.7715
##          Neg Pred Value : 0.8436
##              Prevalence : 0.5000
##          Detection Rate : 0.4310
##    Detection Prevalence : 0.5586
##       Balanced Accuracy : 0.8033
##
##        'Positive' Class : 0
##
```

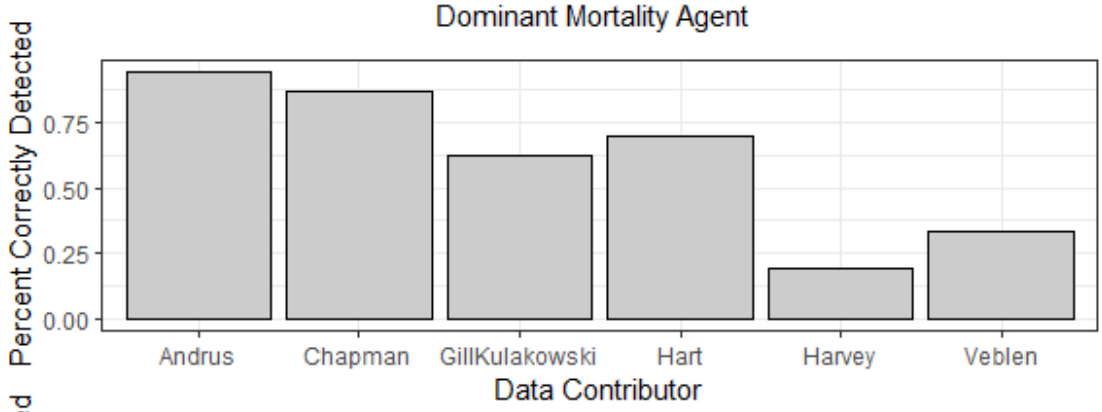
```
## Summarizing model results
preds <- data.frame(observed = obsVals, predicted = predVals,
                    contr, domDist, prevBA, bbSev, sbwSev)
  # Subsetting to just look at those with presence to be able to compare dete
ction
  # based across disturbance severities, initial densities, etc.
preds <- preds[(preds$domDist != ""),]
preds$correct <- as.numeric(preds$predicted == "1")

## Creating some diagnostic plots
# Summarizing by percent mortality
preds$bin <- cut(preds$bbSev, breaks = seq(0, 100, by = 20))
mortSummary <- preds %>%
  group_by(bin) %>%
  summarize(percentCorrect = mean(correct))
# Classification accuracy by percent mortality
a <- ggplot(mortSummary, aes(x = bin, y = percentCorrect)) +
     geom_bar(color = "black", stat = "identity", fill = "grey80") +
     theme_bw() + xlab("Tree Mortality (% BA)") + ylab("Percent Detected") +
     theme(legend.position = "none")

# Summarizing by initial density
preds$bin <- cut(preds$prevBA, breaks = seq(0, 150, by = 30))
mortSummary <- preds %>%
  group_by(bin) %>%
  summarize(percentCorrect = mean(correct))
# Classification accuracy by inital density
b <- ggplot(mortSummary, aes(x = bin, y = percentCorrect)) +
     geom_bar(color = "black", stat = "identity", fill = "grey80") +
     theme_bw() + xlab("Initial Basal Area (sq m/ha)") +
     ylab("Percent Detected") +
     theme(legend.position = "none")

# And summarizing by mortality agent
mortSummary <- preds %>%
  group_by(domDist) %>%
  summarize(percentCorrect = mean(correct))
```

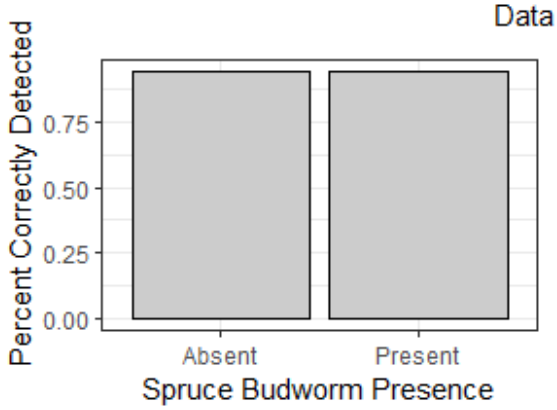
```r
## Removing empty factor level
mortSummary$domDist <- factor(as.character(mortSummary$domDist))

# Classification accuracy by mortality agent
c <- ggplot(mortSummary, aes(x = domDist, y = percentCorrect)) +
      geom_bar(color = "black", stat = "identity", fill = "grey80") +
      theme_bw() + xlab("Dominant Mortality Agent") + ylab("Percent Correctly
Detected") +
      theme(legend.position = "none")

# Summarizing by contributor
mortSummary <- preds %>%
  group_by(contr) %>%
  summarize(percentCorrect = mean(correct))

# Classification accuracy by mortality agent
d <- ggplot(mortSummary, aes(x = contr, y = percentCorrect)) +
      geom_bar(color = "black", stat = "identity", fill = "grey80") +
      theme_bw() + xlab("Data Contributor") + ylab("Percent Correctly Detecte
d") +
      theme(legend.position = "none")

# And summarizing sbw by presence/absence
preds <- preds[!is.na(preds$sbwSev),]
preds$sbwPres <- as.factor(ifelse(preds$sbwSev == 0, "Absent", "Present"))
mortSummary <- preds %>%
  group_by(sbwPres) %>%
  summarize(percentCorrect = mean(correct), percentMortality = mean(bbSev),
            seMort = sd(bbSev)/sqrt(n()))

# Potential effect of spruce budworm on classification accuracy
e1 <- ggplot(mortSummary, aes(x = sbwPres, y = percentCorrect)) +
        geom_bar(color = "black", stat = "identity", fill = "grey80") +
        theme_bw() + xlab("Spruce Budworm Presence") + ylab("Percent Correctl
y Detected") +
        theme(legend.position = "none")
# Next to plot with
e2 <- ggplot(mortSummary, aes(x = sbwPres, y = percentMortality)) +
        geom_bar(color = "black", stat = "identity", fill = "grey80") +
        theme_bw() + xlab("Spruce Budworm Presence") + ylab("BB Mortality (%
BA)") +
        theme(legend.position = "none") +
        geom_errorbar(aes(ymin=percentMortality-seMort, ymax=percentMortality
+seMort),
                      width=.2, position=position_dodge(.9))
e <- cowplot::plot_grid(e1, e2, nrow = 1)

cowplot::plot_grid(a, b, c, d, e, nrow = 5)
```

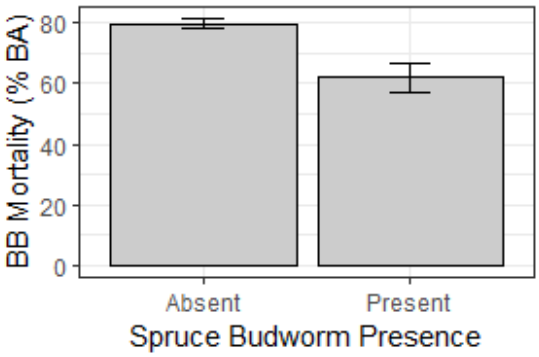

Percent Detected
1.00
0.75
0.50
0.25
0.00
(0,20]
(20,40]
(40,60]
(60,80]
(80,100]
Tree Mortality (% BA)


Percent Detected
1.00
0.75
0.50
0.25
0.00
(0,30]
(30,60]
(60,90]
(90,120]
(120,150]
Initial Basal Area (sq m/ha)


Percent Correctly Detected
0.8
0.6
0.4
0.2
0.0
MPB
SB
SFD
Dominant Mortality Agent


Percent Correctly Detected
0.75
0.50
0.25
0.00
Andrus
Chapman
GillKulakowski
Hart
Harvey
Veblen
Data Contributor


Percent Correctly Detected
0.75
0.50
0.25
0.00
Absent
Present
Spruce Budworm Presence

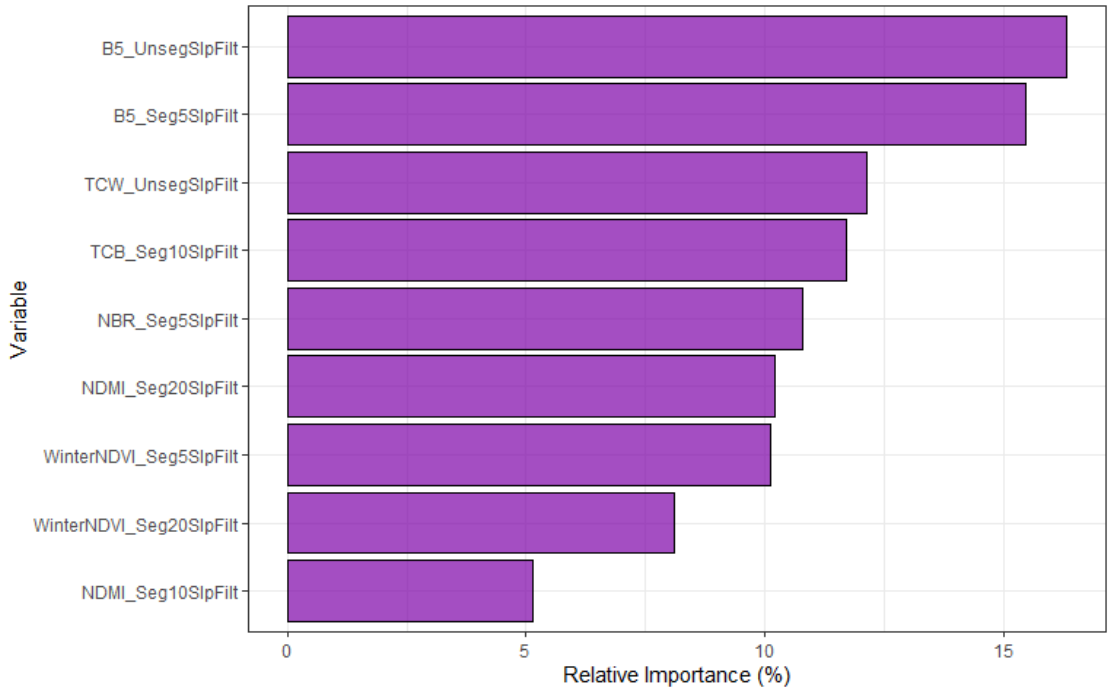

BB Mortality (% BA)
80
60
40
20
0
Absent
Present
Spruce Budworm Presence

We can see in the figures above that it is more difficult to identify the presence of disturbance in plots that had lower-severity bark beetle activity. This is reflected both in the first plot that shows correct classification by bins of mortality, and by the last figure that shows accuracy by mortality agent. The model is performing less well on plots that were primarily affected by SFD, but these tended to be in the lower-severity range, so I don't think it is too surprising that they are being classified as "no disturbance" in the disturbance presence/absence layer. The trends by initial basal area show that there doesn't appear to be a strong classification bias based on this, which is good. Interestingly, plots with spruce budworm (in addition to bark beetle attack) are more likely to be detected than those without defoliation, even though tree mortality is lower on average.

Variable importance and partial dependence plots for the presence/absence model

The following chunks plot variable importance and partial dependence for each of the spectral indices and bands that were included in the final Random Forest model of bark beetle presence.

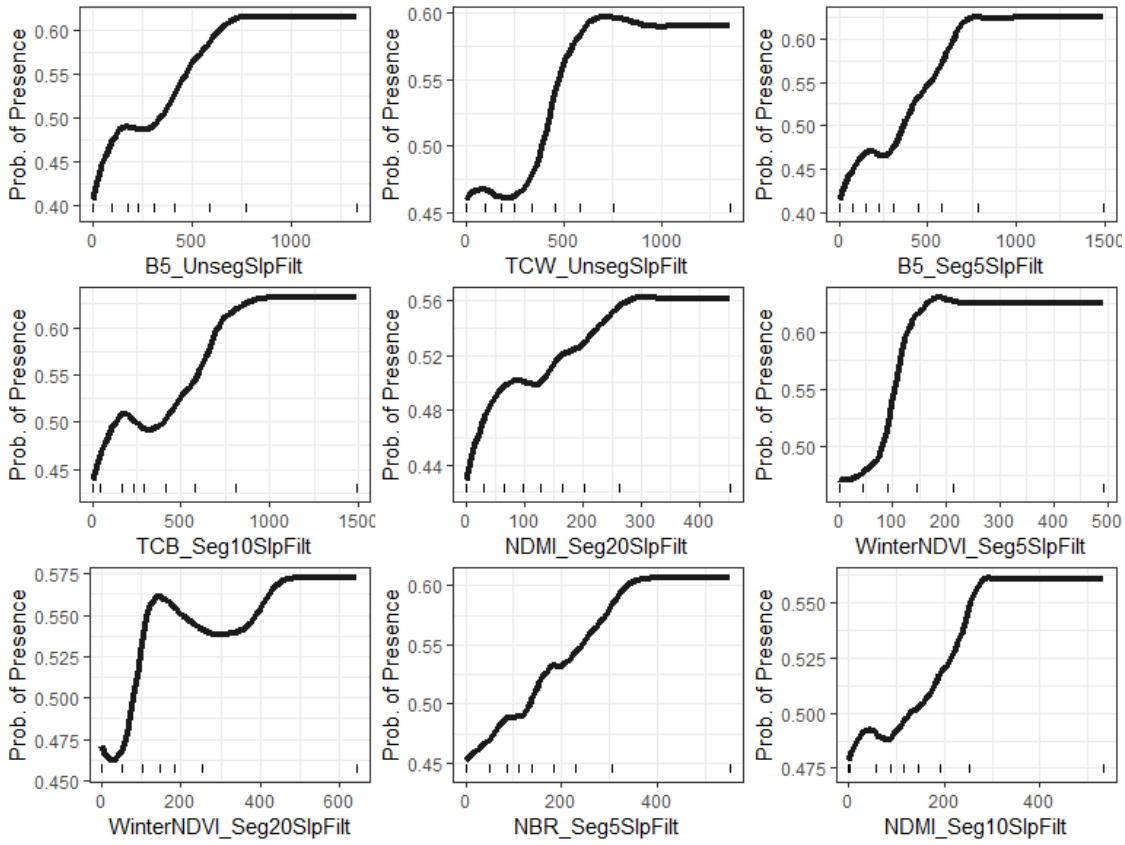
```
## Plotting variable importance
# First, get vimp, order by value, and scale to 0-100
vimp <- stack(presMod$variable.importance)
vimp$ind <- factor(vimp$ind, levels = vimp$ind[order(vimp$values, decreasing
= F)])
vimp$values <- vimp$values/sum(vimp$values)*100

# Then plotting variable importance
(vimp1 <- ggplot(vimp, aes(x = ind, y = values)) +
  geom_bar(stat = "identity", color = "black", fill = bbColor, alpha = 0.7) +
  coord_flip() + theme_bw() + xlab("Variable") + ylab("Relative Importance (%
)") +
  theme(legend.position = "none"))
```



Many of the bands/indices included in the final RF model target the shortwave infrared portion of the spectrum (i.e., bands 5 and 7, NBR, NDMI, tasseled cap indices). This aligns with prior literature on using Landsat data to detect tree mortality. NDVI (which only includes red and near-infrared) in snow-on conditions in the winter seems helpful too - by focusing on the winter period, the signal of conifer mortality is enhanced and aspen/understory vegetation are minimized.

```
## Re-fitting probability forest for easier visualization of partial plots
presModProb <- ranger(formula = DisturbanceType~., data = modData,
                      importance = "permutation", num.trees = 1000,
                      min.node.size = optMinNode, splitrule = optSplit,
                      mtry = optMtry, seed = 25, probability = T)

## And plotting partial dependence. Re-fitting this with probability forest t
o get
  # class probabilities.
partialVals <- lapply(vimp$ind, FUN = function(x, data = modData){
  ## Calculating partial values
  allData <- pdp::partial(presModProb, pred.var = as.character(x),
                type = "classification", which.class = "1", prob = T)
  colnames(allData)[2] <- c("BBProbability")

  ## Getting focal column from data frame
  focalVec <- as.vector(data[,colnames(data) == x])

  ## Plotting
  p <- ggplot(allData, aes(x = allData[,1])) +
      geom_smooth(aes(y = BBProbability), color = "grey10", lwd = 1.5,
```

```
                    se = F, span = 0.25) +
      xlab(as.character(x)) + ylab("Prob. of Presence") + theme_bw()

  # Adding deciles similar to rug plot but with less overplotting. First gett
ing
    # existing range, then expanding down, then calculating location of ticks
.
    ylims <- ggplot_build(p)$layout$panel_scales_y[[1]]$range$range
    p <- p + ylim(c(ylims[1] - abs((ylims[2]-ylims[1]) * 0.07), ylims[2]))
    ystart <- ylims[1]- abs((ylims[2]-ylims[1]) * 0.07)
    yend <- ylims[1]- abs((ylims[2]-ylims[1]) * 0.02)

    for(i in 1:11){
      dec <- quantile(focalVec, probs = c((i-1) *0.1))
      p <- p+annotate("segment", y = ystart, yend = yend, x = dec,
                      xend = dec, lwd = 0.7)
    }

  ## Returning plot
  return(p)
})

## Merging them all
sjPlot::plot_grid(partialVals, margin = c(0.1,0.1,0.1,0.1))
```



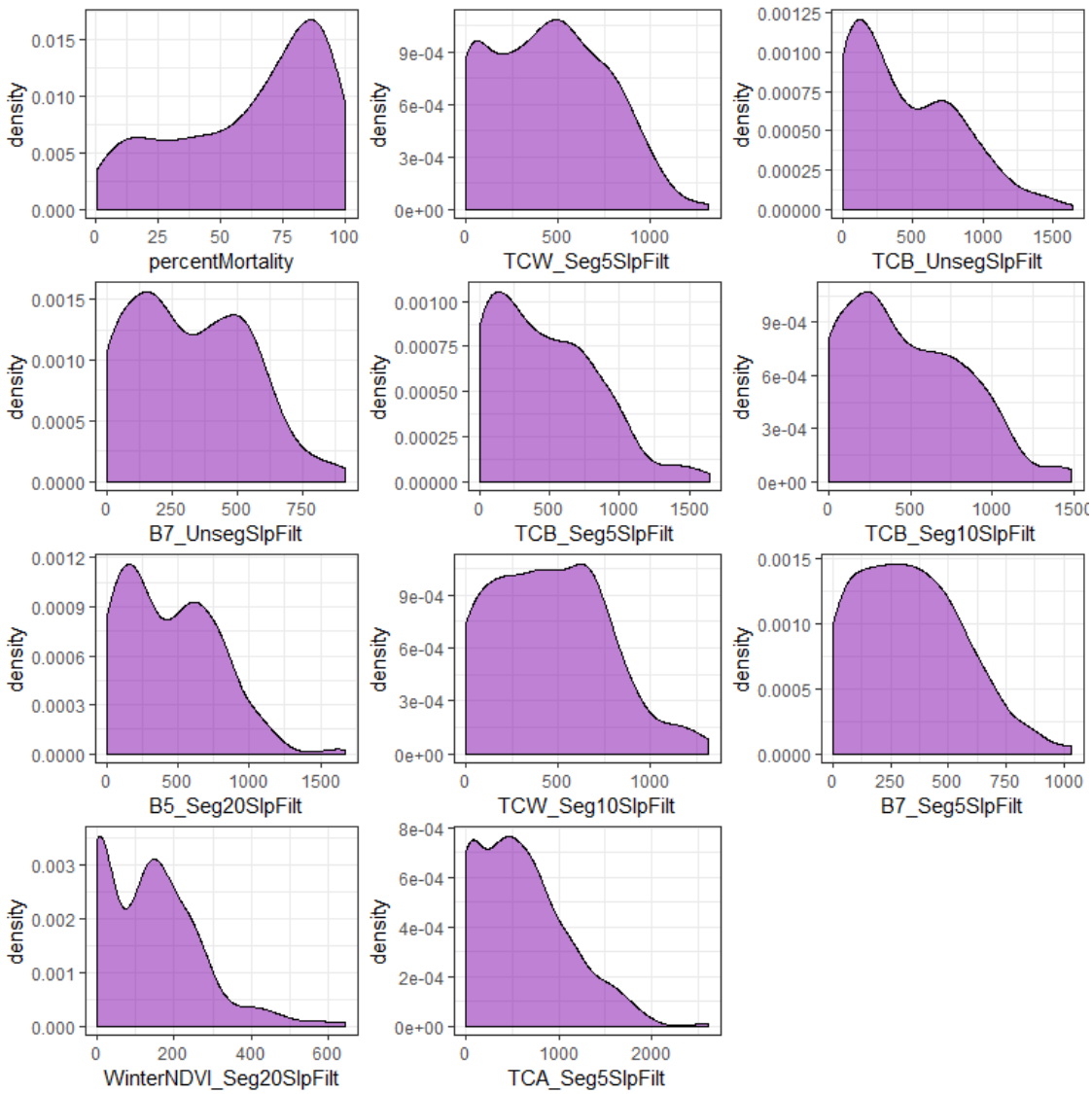
Partial dependence plots above all show logical trends. Basically, the probability of bark beetle disturbance increases with the amount of spectral change from 1997-2019 in each of these bands/indices.

The RF model of disturbance severity

Plotting distributions of the different variables in the severity model

The plots below show the sampled distribution of the response variable (percentMortality) and each of the selected predictors from LandTrendr in the final model of disturbance severity. As in the model above, values of each predictor are oriented such that higher numbers are indicative of greater spectral declines indicative of vegetation loss.

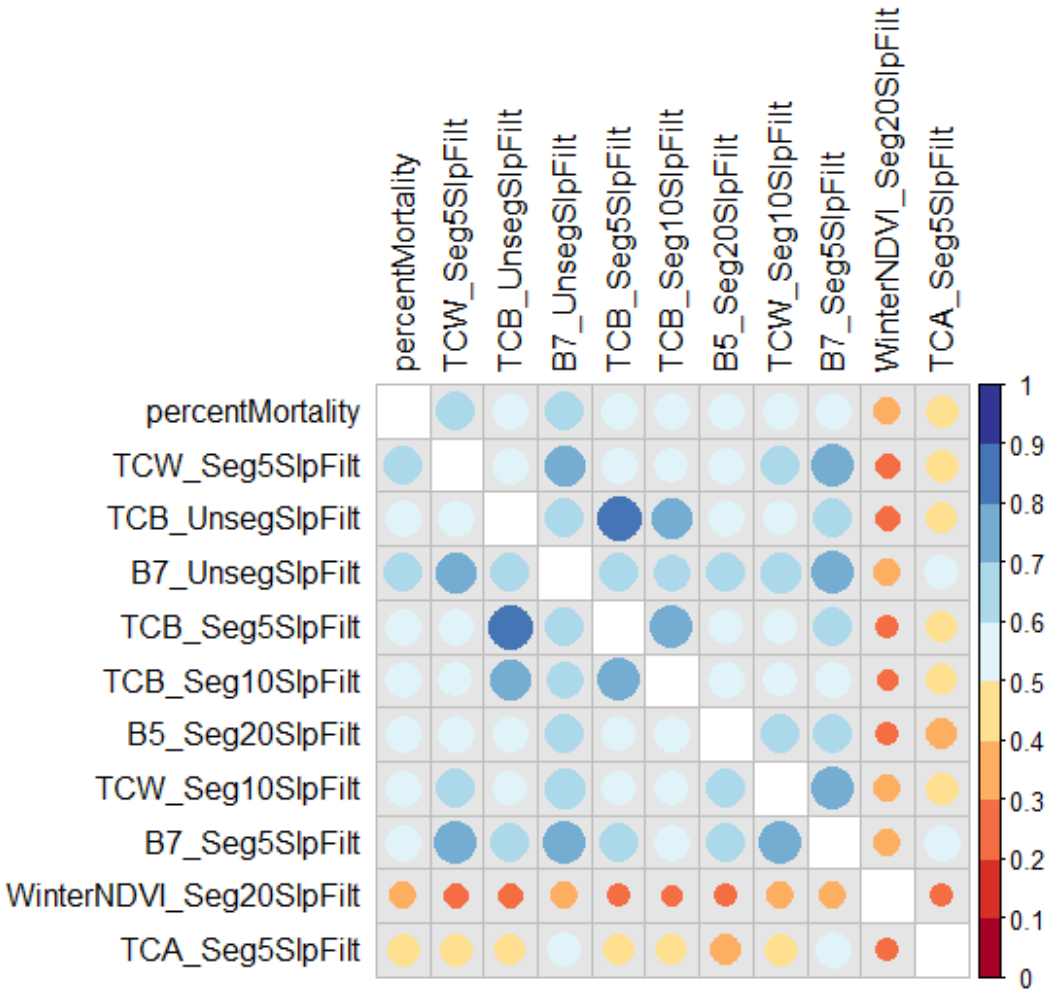
```
## Getting indices of columns
start <- which(colnames(fieldData) == "percentMortality")
end <- ncol(fieldData)

## Making plots
densPlots <- lapply(colnames(fieldData)[start:end],
                    FUN = function(x, data = fieldData){

  ## Getting focal column from data frame
  focalData <- data.frame(data[,colnames(data) == x])

  ## Plotting
  p <- ggplot(focalData, aes(x = focalData[,1])) +
    geom_density(fill = bbColor, alpha = 0.5) +  xlab(as.character(x)) +
    ylab("density") + theme_bw()

  ## Returning plot
  return(p)
})

## Putting list of plots together
sjPlot::plot_grid(densPlots, margin = c(0.1,0.1,0.1,0.1))
```



And making a correlation plot of the response and each predictor

The plot below shows the correlations among each of the predictors and the response variable. As in a similar plot above, all correlations are positive, with lower values in red and higher values in blue.

```
## Creating pairwise correlation matrix
corrMat <- cor(fieldData[start:end])

## Color ramp - hacky way to get the function below to work is to repeat twic
e
pal <- brewer.pal(10, name = "RdYlBu"); pal <- c(pal, pal)

## Making correlation plot
corrplot(corrMat, method = "circle", diag = F, cl.lim = c(0, 1),
         tl.col = "black", col = pal, bg = "grey90")
```



Tuning and fitting the Random Forest model of percent basal area loss

The code chunk below performs recursive feature elimination, tunes hyperparameters (i.e., number of variables to split at each node, splitting rule, and minimum node size), and fits the final Random Forest model of disturbance severity.

```
## Getting subset of variables with predictors and response
modData <- fieldData[,c(start:end)]

## Testing for multicollinearity of predictors
multi.collinear(modData[,-1])

## NULL

## Creating CV Folds
set.seed(25)
folds <- createFolds(modData$percentMortality, k = nfolds, list = TRUE)
```

```r
## Setting parameters for recursive feature elimination (i.e., backward selec
tion)
  # First establishing seeds for multi-core
seeds <- setSeeds(numbers = nfolds, tunes = ncol(modData)-1, seed = 118)
  # And setting parameters
controls <- rfeControl(method = "cv", number = nfolds,
                       index = folds, seeds = seeds, functions = rfFuncs)
  # Initiating parallel processing and setting seed
cl <- makeCluster(detectCores()-1)
registerDoParallel(cl)
set.seed(118)

## Running recursive feature elimination
caretRFE <- rfe(y = modData$percentMortality,
                x = within(modData, rm(percentMortality)),
                rfeControl = controls,
                sizes = ncol(modData)-1)
stopCluster(cl)
#caretRFE ## Including all terms is the best option here

# Set hyperparameters for tuning
controls <- trainControl(method = "cv", number = nfolds,
                         index = folds,
                         savePredictions = "all")
tgrid <- expand.grid(
  mtry = 2:5,
  splitrule = c("variance", "maxstat"),
  min.node.size = c(1, 5, 10, 25, 50, 100)
)

# Running model on tuning grid
set.seed(92)
cl <- makeCluster(detectCores()-1)
registerDoParallel(cl)
model_caret <- train(percentMortality~., data = modData,
                     method = "ranger",
                     trControl = controls,
                     tuneGrid = tgrid,
                     importance = "permutation", num.trees = 1000)

## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =
## trainInfo, : There were missing values in resampled performance measures.

stopCluster(cl)

## Getting optimal hyperparameters from cross-validation above
optMtry <- as.numeric(model_caret$bestTune[1,1])
optSplit <- as.character(model_caret$bestTune[1,2])
optMinNode <- as.numeric(model_caret$bestTune[1,3])
```

```r
## Getting other variables to track in crossvalidation
mortAgent <- fieldData$domDisturb; baPre <- fieldData$preBA_HA
sbwTrees <- fieldData$sbw_Trees; contrib <- fieldData$contributor

## Performing 10-fold cross-validation of model
predVals <- c(); obsVals <- c(); domDist <- c(); prevBA <- c(); contr <- c();
sbwSev <- c()

## Getting seeds for crossvalidation
  # Creating separate folds for accuracy assessment
set.seed(924)
folds <- createFolds(modData$percentMortality, k = nfolds, list = TRUE)
set.seed(94)
cvSeeds <- sample(1:1000, nfolds)

for(i in 1:nfolds){
  ## Subsetting data
  testData <- modData[folds[[i]],] ## Subsetting to test set
  trainData <- modData[-folds[[i]],] ## Subsetting to training set

  ## Getting other tracked values - contributor, previous BA, and sbw defolia
tion
  domDist <- c(domDist, as.character(mortAgent[folds[[i]]]))
  prevBA <- c(prevBA, baPre[folds[[i]]])
  sbwSev <- c(sbwSev, sbwTrees[folds[[i]]])
  contr <- c(contr, as.character(contrib[folds[[i]]]))

  ## Fitting model to training data and predicting to test set
  mod <- ranger(formula = percentMortality~., data = trainData,
                importance = "permutation",
               num.trees = 1000, min.node.size = optMinNode,
               mtry = optMtry, splitrule = optSplit, seed = cvSeeds[i])

  ## And getting those values
  obsVals <- c(obsVals, testData$percentMortality)
  predVals <- c(predVals, predict(mod, data = testData)$predictions)
}

## Developing final RF model of disturbance severity with selected parameters
  # and full dataset
(severMod <- ranger(formula = percentMortality~., data = modData,
                importance = "permutation",num.trees = 1000,
                min.node.size = optMinNode, splitrule = optSplit,
               mtry = optMtry, seed = 202))

## Ranger result
##
## Call:
```

```
##  ranger(formula = percentMortality ~ ., data = modData, importance = "perm
utation",      num.trees = 1000, min.node.size = optMinNode, splitrule = optS
plit,      mtry = optMtry, seed = 202)
##
## Type:                             Regression
## Number of trees:                  1000
## Sample size:                      239
## Number of independent variables:  10
## Mtry:                             2
## Target node size:                 1
## Variable importance mode:         permutation
## Splitrule:                        maxstat
## OOB prediction error (MSE):       274.2722
## R squared (OOB):                  0.6794364

# saveRDS(severMod, here("AnalysisOutputs", "BB_SeverityMod.rds"))
```

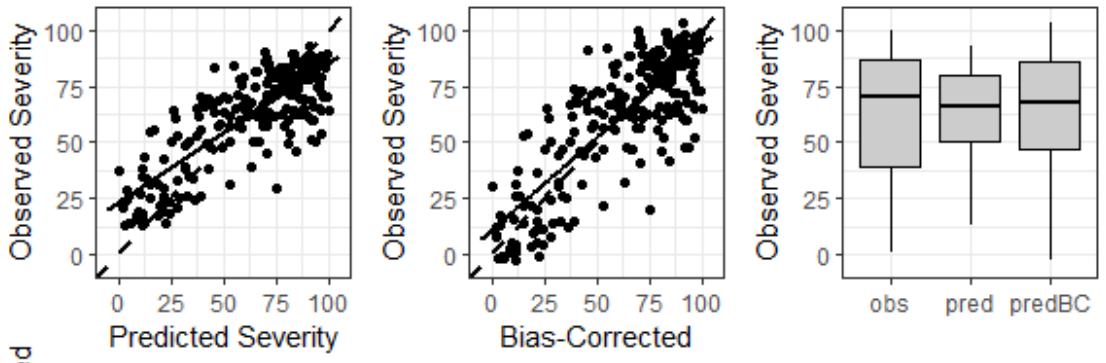
Summarizing results of the model of percent basal area loss

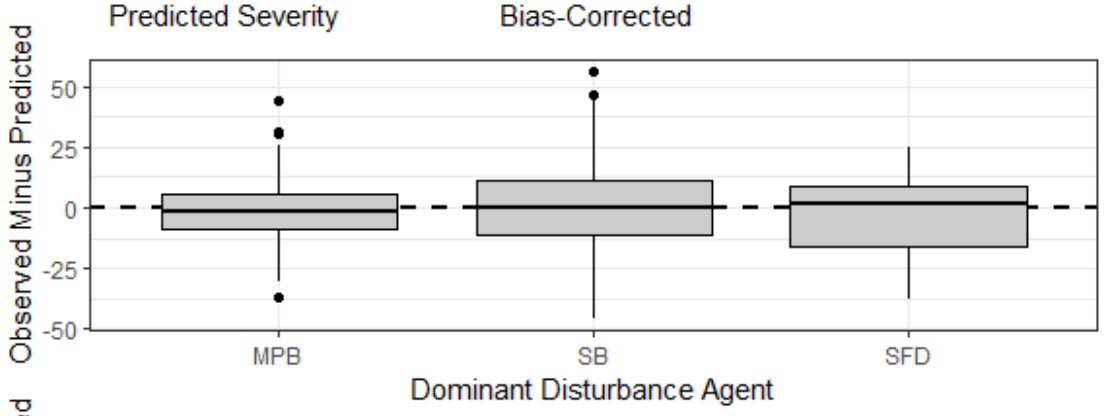
The model of bark beetle severity is performing pretty well in predicting the percentage of basal area that died c. 1997-2019 (R-squared in the OOB sample of 0.68) The code chunk below prints the accuracy of the model in 10-fold cross-validation and makes diagnostic plots that describe the model predictions by contributor, mortality agent, across a range of initial densities and defolation severities.

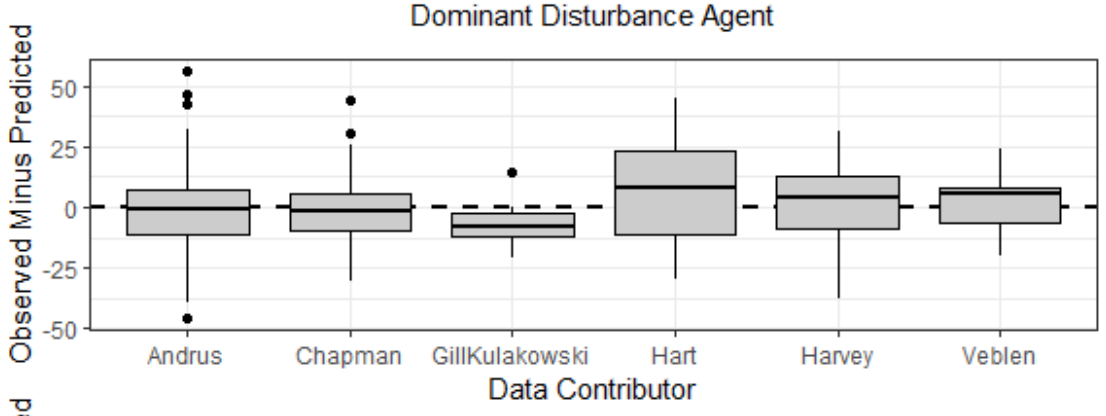
```
## Creating function to calculate RMSE
rmse <- function(act, pred){return(sqrt(mean((act - pred)^2)))}
# cat("Severity model has an cross-validated RMSE of: ")
# rmse(obsVals, predVals)
# cat(" percent\nand a cross-validated R-squared of: ")
# cor(obsVals, predVals)^2

## Plotting model results
preds <- data.frame(observed = obsVals, predicted = predVals,
                    contr, domDist, prevBA, sbwSev)

## Getting mean and sd of obs and predicted in crossvalidation, for bias corr
ection
# muRF <- mean(predVals); sigmaRF <- sd(predVals)
# muObs <- mean(obsVals); sigmaObs <- sd(obsVals)

## Transforming using variance matching - mean and sd of observed and predict
ed in cross-validation
  # this was the method we ended up using
preds$predCorrected <- (((preds$predicted - 61.65879)/22.06188) * 29.25053) +
62.07586

  # Updated fitting statistics
cat("Severity model has an cross-validated RMSE of: ")
```

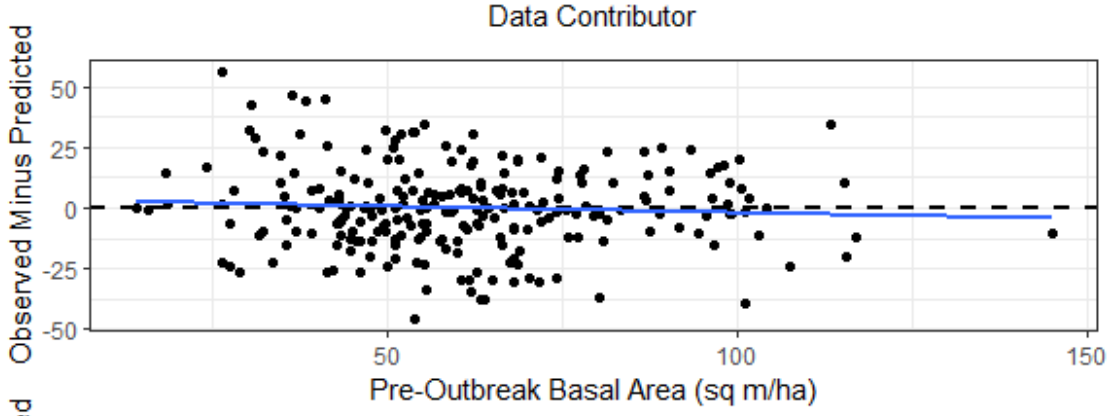
```r
## Severity model has an cross-validated RMSE of:

rmse(preds$observed, preds$predCorrected)

## [1] 17.31348

cat(" percent\nand a cross-validated R-squared of: ")

##  percent
## and a cross-validated R-squared of:

cor(preds$observed, preds$predCorrected)^2

## [1] 0.6791231

## Creating some diagnostic plots
  # Observed vs. predicted and 1:1 line, after bias correction
a1 <- ggplot(preds, aes(y = predicted, x = observed)) +
      geom_point() +  geom_abline(intercept = 0, slope = 1, lty = 2, lwd = 1)
+
      geom_smooth(method = "lm", se = F, lwd = 1, lty = 1, fullrange = T, col
or = "black") +
      theme_bw() + xlab("Predicted Severity") + ylab("Observed Severity") +
      xlim(c(-5,105)) + ylim(c(-5,105)) + theme(legend.position = "none")

a2 <- ggplot(preds, aes(y = predCorrected, x = observed)) +
      geom_point() +  geom_abline(intercept = 0, slope = 1, lty = 2, lwd = 1)
+
      geom_smooth(method = "lm", se = F, lwd = 1, lty = 1, fullrange = T, col
or = "black") +
      theme_bw() + xlab("Bias-Corrected") + ylab("Observed Severity") +
      xlim(c(-5,105)) + ylim(c(-5,105)) + theme(legend.position = "none")
  # Merging to single long-form DF
predLong <- preds %>%
  select(c(observed, predicted, predCorrected))%>%
  rename(obs = observed, pred = predicted, predBC = predCorrected) %>%
  gather()
predLong$key <- factor(predLong$key, levels = c("obs", "pred", "predBC"))

  # Observed vs. predicted (following bias correction) and 1:1 line
a3 <- ggplot(predLong, aes(y = value, x = key)) +
      geom_boxplot(color = "black", fill = "grey80") +
  theme_bw() + xlab(" ") + ylab("Observed Severity") +
      ylim(c(-5,105)) + theme(legend.position = "none")
a <- cowplot::plot_grid(a1, a2, a3, nrow = 1)

  # Residuals across different disturbance agents
b <- ggplot(preds, aes(x = domDist, y = observed-predCorrected)) +
      geom_hline(yintercept = 0, lty = 2, lwd = 1) +
      geom_boxplot(color = "black", fill = "grey80") +
      theme_bw() + ylab("Observed Minus Predicted") +
```

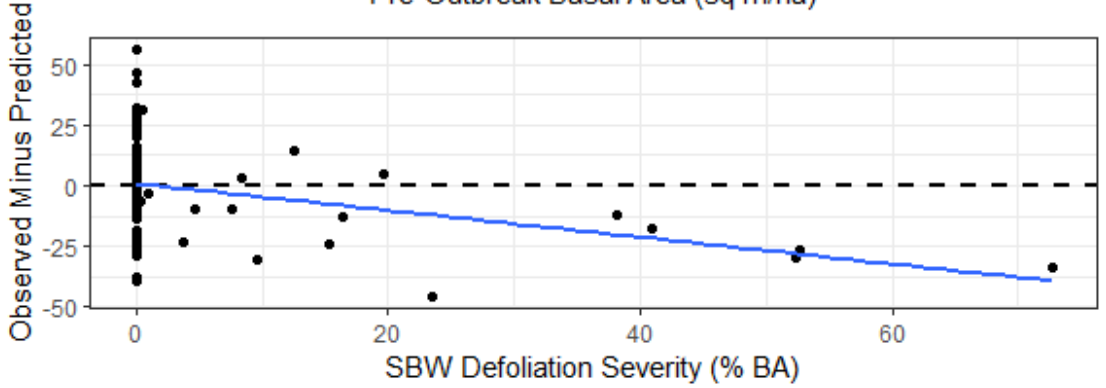
```
      xlab("Dominant Disturbance Agent") + theme(legend.position = "none")

  # Residuals across different disturbance agents
c <- ggplot(preds, aes(x = contr, y = observed-predCorrected)) +
      geom_hline(yintercept = 0, lty = 2, lwd = 1) +
      geom_boxplot(color = "black", fill = "grey80") +
      theme_bw() + ylab("Observed Minus Predicted") +
      xlab("Data Contributor") + theme(legend.position = "none")

  # Residuals by initial density
d <- ggplot(preds, aes(x = prevBA, y = observed-predCorrected)) +
      geom_hline(yintercept = 0, lty = 2, lwd = 1) + geom_point() +
      geom_smooth(method = "gam", se = F) + theme_bw() + ylab("Observed Minus
Predicted") +
      xlab("Pre-Outbreak Basal Area (sq m/ha)")

  # Residuals by sbw defoliation
e <- ggplot(preds, aes(x = sbwSev, y = observed-predCorrected)) +
      geom_hline(yintercept = 0, lty = 2, lwd = 1) + geom_point() +
      geom_smooth(method = "gam", se = F) + theme_bw() + ylab("Observed Minus
Predicted") +
      xlab("SBW Defoliation Severity (% BA)")

cowplot::plot_grid(a, b, c, d, e, nrow = 5)
```


Observed Severity
Predicted Severity
Bias-Corrected
obs
pred
predBC


Observed Minus Predicted
MPB
SB
SFD
Dominant Disturbance Agent


Observed Minus Predicted
Andrus
Chapman
GillKulakowski
Hart
Harvey
Veblen
Data Contributor


Observed Minus Predicted
Pre-Outbreak Basal Area (sq m/ha)

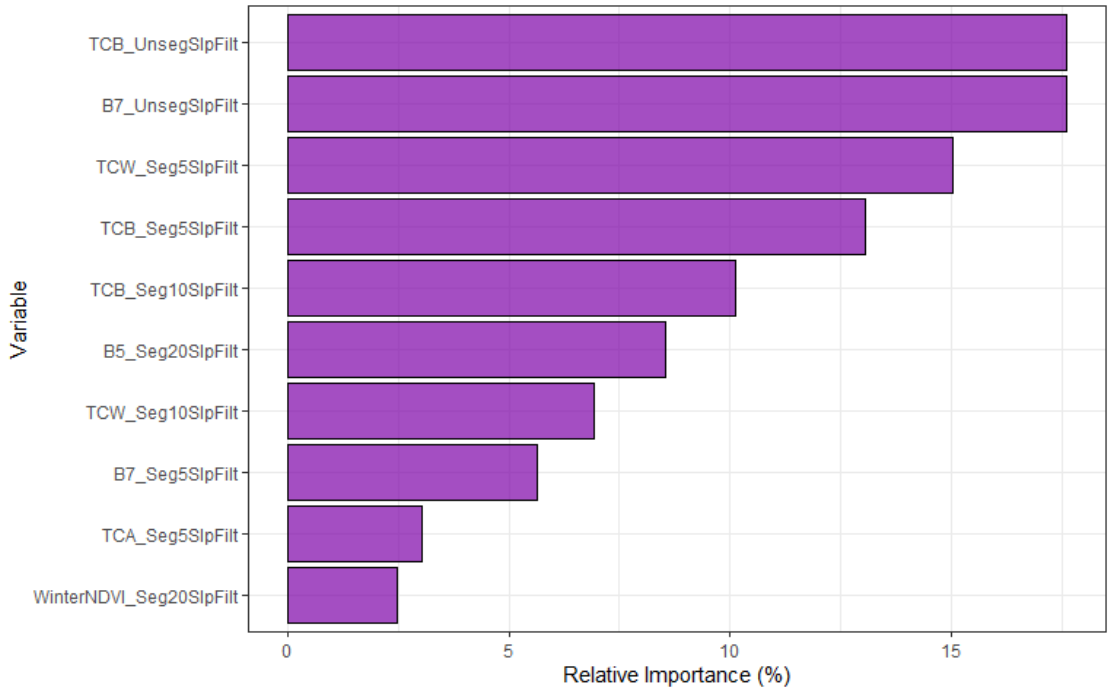

Observed Minus Predicted
SBW Defoliation Severity (% BA)

As shown in the figures and summaries above, the RF model of severity is predicting well into new areas (i.e., the cross-validated sample) with a fairly low RMSE of ~17%BA and a high R-squared for observed vs. predicted of ~0.7. The scatterplot shows the observed to predicted relationship with point shapes and colors based on dominant mortality agent. The relationship showed bias at low and high values and the RF-predicted values did not span the full range of the data (a common problem in Random Forest regression). We addressed this in the final data layers through bias correction of predicted values using a z-score standardization based on 10-fold cross-validation (i.e., adjusting predictions to match the mean and standard deviation of observed values). The results of this are shown in the boxplots, where predicted values (following bias correction) more closely match the range and distribution of the observed data. There doesn't appear to be much of a bias in predictions by mortality agent (shown in boxplots), indicating that the model is doing well at predicting in different contexts. Similarly, there are no notable biases by data contributor, which is important given the slightly different plot types and field protocols. There also doesn't appear to be a bias in severity predictions by initial density. As with fire severity metrics such as dNBR, we might expect that the magnitude of spectral change could be related to the initial density of the forest, but it seems like that isn't too much of an issue in the subalpine stands here. In part, this could be because many of the plots were pretty dense before outbreak occurrence, but there were some more open ones too. SBW defoliation seems to only have a bit of an effect on predictions of severity, where we could be overpredicting the severity of bark beetle attack by ~20% in stands with 50% defoliation by spruce budworm. Though field data to assess this relationship is limited in areas with higher defoliation.

Variable importance and partial dependence plots for the severity model

The following chunks plot variable importance and partial dependence for each of the spectral indices and bands that were included in the final Random Forest model.

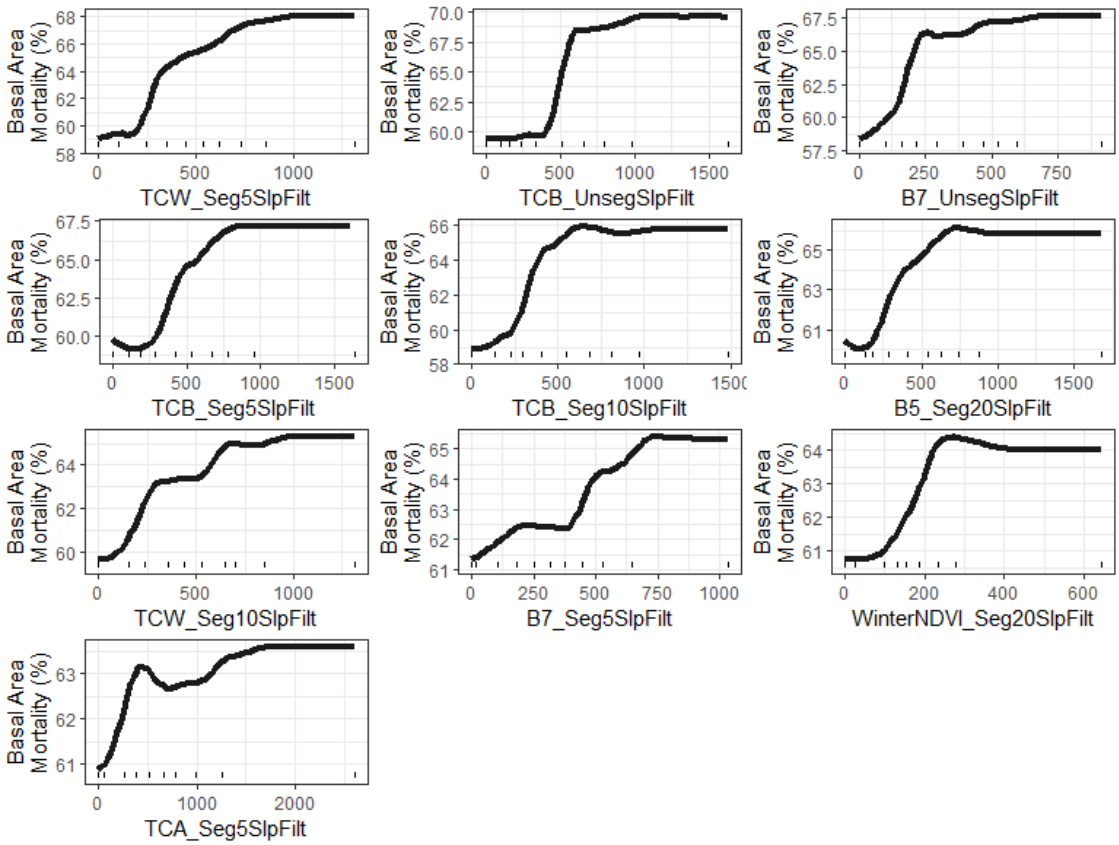
```
## Plotting variable importance
# First, get vimp, order by value, and scale to 0-100
vimp <- stack(severMod$variable.importance)
vimp$ind <- factor(vimp$ind, levels = vimp$ind[order(vimp$values, decreasing
= F)])
vimp$values <- vimp$values/sum(vimp$values)*100

# Then plotting variable importance
(vimp2 <- ggplot(vimp, aes(x = ind, y = values)) +
  geom_bar(stat = "identity", color = "black", fill = bbColor, alpha = 0.7) +
  coord_flip() + theme_bw() + xlab("Variable") + ylab("Relative Importance (%
)") +
  theme(legend.position = "none"))
```



Again, as with the presence/absence model, bands and indices that incorporate the shortwave infrared portion of the spectrum were most often retained by VSURF and appear to be most influential predictors in the final model. Winter NDVI seems a little less helpful here than in the presence/absence model.

```
## Creating partial dependence plots
partialVals <- lapply(vimp$ind, FUN = function(x, data = modData){
  ## Calculating partial values
  response <- pdp::partial(severMod, pred.var = as.character(x),
                   type = "regression")

  ## Getting focal column from data frame
  focalVec <- as.vector(data[,colnames(data) == x])

  ## Plotting
  p <- ggplot(response, aes(x = response[,1])) +
    geom_smooth(aes(y = response[,2]), color = "grey10", lwd = 1.5, se = F, s
pan = 0.25) +
    xlab(as.character(x)) + ylab("Basal Area\nMortality (%)") + theme_bw()

  # Adding deciles similar to rug plot but with less overplotting. First gett
ing
    # existing range, then expanding down, then calculating location of ticks
.
  ylims <- ggplot_build(p)$layout$panel_scales_y[[1]]$range$range
  p <- p + ylim(c(ylims[1] - abs((ylims[2]-ylims[1]) * 0.07), ylims[2]))
  ystart <- ylims[1]- abs((ylims[2]-ylims[1]) * 0.07)
  yend <- ylims[1]- abs((ylims[2]-ylims[1]) * 0.02)
```

```
  for(i in 1:11){
    dec <- quantile(focalVec, probs = c((i-1) *0.1))
    p <- p+annotate("segment", y = ystart, yend = yend, x = dec, xend = dec,
lwd = 0.7)
  }

  ## Returning plot
  return(p)
})

## And creating a plot with them all
sjPlot::plot_grid(partialVals, margin = c(0.1,0.1,0.1,0.1))
```



Again, all of the partial dependence plots look good and give reasonable relationships. As the magnitude of change increases for each band/spectral index, the predicted outbreak severity increases. There are interesting non-linear relationships for some of these things.