WSVHA Revisions (New Approved Masks and VRI added)

Cabin-GIS

28/06/2022

Contents

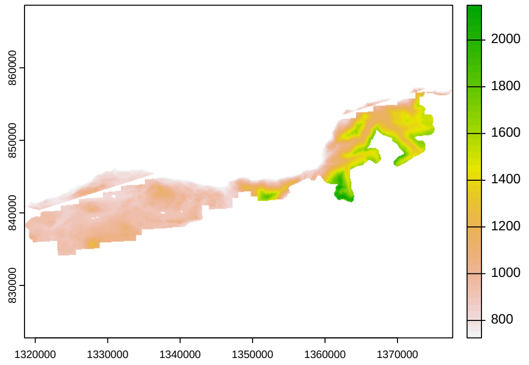
Action:

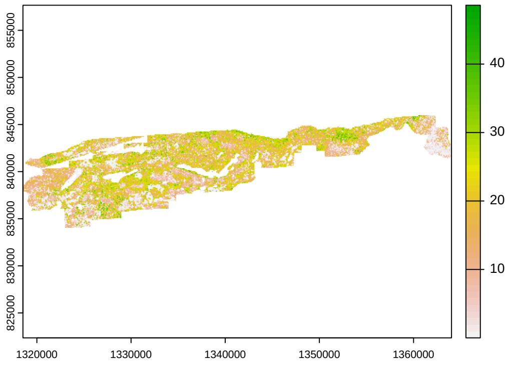
The following report includes revisions to WSVHA outptus for Quesnel Lake and other All-Species Areas Fd-stands and nonFd-stands derived with newly approved masking and VRI data layers shared by LQ.

1. Quesnel Lake: Includes areas for all species (Fd and non-Fd stands), and developed cutblocks for validation purposes

1.1. Quesnel Lake: Import DEM & CHM

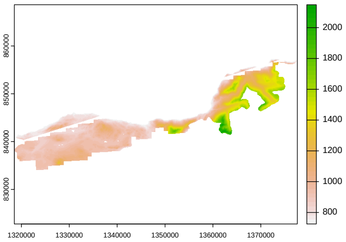
```
filez_be_quesnel = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/BareEarth", full.nar
elev_raster_list_quesnel <- lapply(filez_be_quesnel, raster)
elev_raster_quesnel = do.call(merge, c(elev_raster_list_quesnel, tolerance = 1))
elev_rast_quesnel = terra::rast(elev_raster_quesnel)
terra::crs(elev_rast_quesnel) = "epsg:3005"
elev_rast_quesnel = terra::aggregate(elev_rast_quesnel, fact = 100, fun = mean)
writeRaster(elev_rast_quesnel, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/elev_ras
elev_raster_quesnel = raster::raster("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/elev_raster_
elev_rast_quesnel = terra::rast(elev_raster_quesnel)

filez_vh_quesnel = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/VegHt", full.names =
lead_htop_raster_list_quesnel <- lapply(filez_vh_quesnel, raster)
lead_htop_raster_quesnel = do.call(merge, c(lead_htop_raster_list_quesnel, tolerance = 1))
lead_htop_rast_quesnel = terra::rast(lead_htop_raster_quesnel)
terra::crs(lead_htop_rast_quesnel) = "epsg:3005"
lead_htop_rast_quesnel = terra::aggregate(lead_htop_rast_quesnel, fact = 100, fun = mean)
lead_htop_rast_quesnel = terra::resample(lead_htop_rast_quesnel, elev_rast_quesnel)
writeRaster(lead_htop_rast_quesnel, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/lea
lead_htop_raster_quesnel = raster::raster("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/lead_ht
lead_htop_rast_quesnel = terra::rast(lead_htop_raster_quesnel)
```

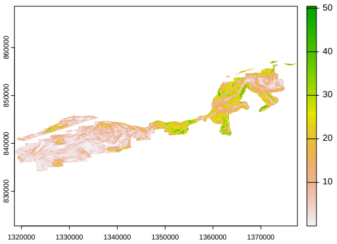


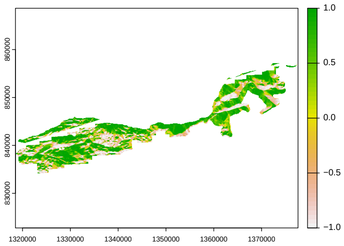


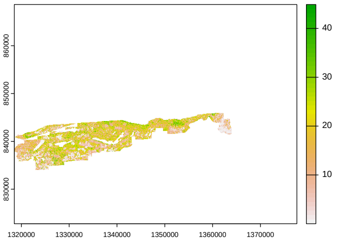
1.2. Quesnel Lake: Derive Terrain Rasters

```
slope_rast_quesnel = terra::terrain(elev_rast_quesnel, v="slope", unit="degrees", neighbors=8)
aspect_rast_quesnel = terra::terrain(elev_rast_quesnel, v="aspect", unit="degrees", neighbors=8)
asp_cos_rast_quesnel = cos((aspect_rast_quesnel*pi)/180)
asp_sin_rast_quesnel = sin((aspect_rast_quesnel*pi)/180)
writeRaster(elev_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked-co
writeRaster(slope_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked-c
writeRaster(asp_cos_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked
writeRaster(asp_sin_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked
writeRaster(lead_htop_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmask
```



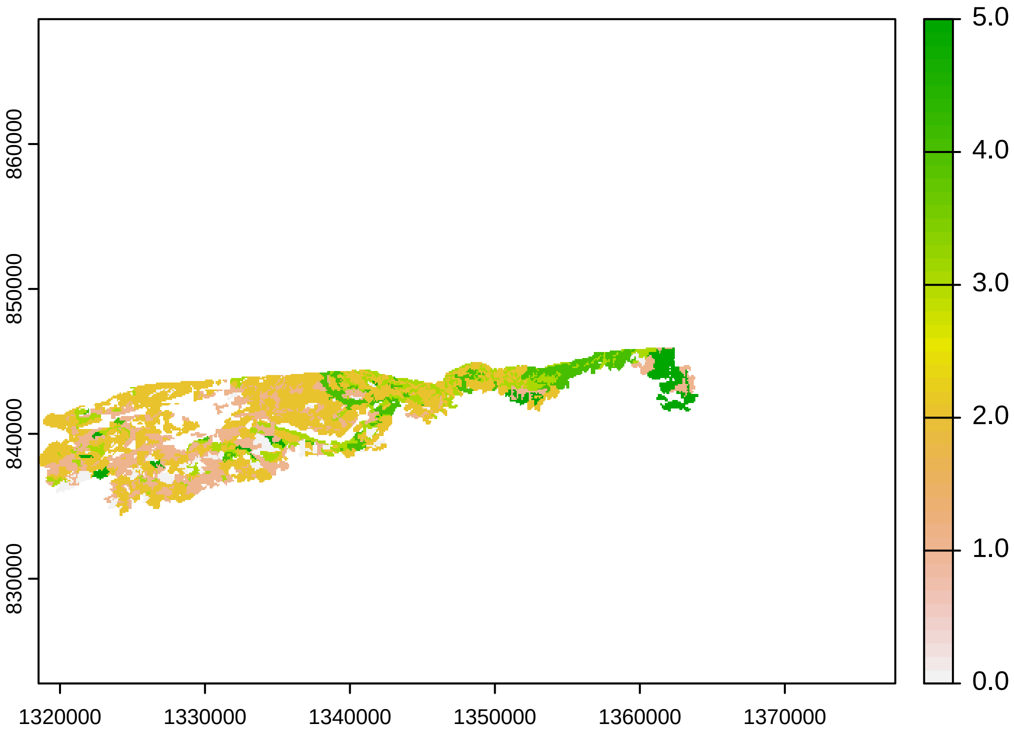






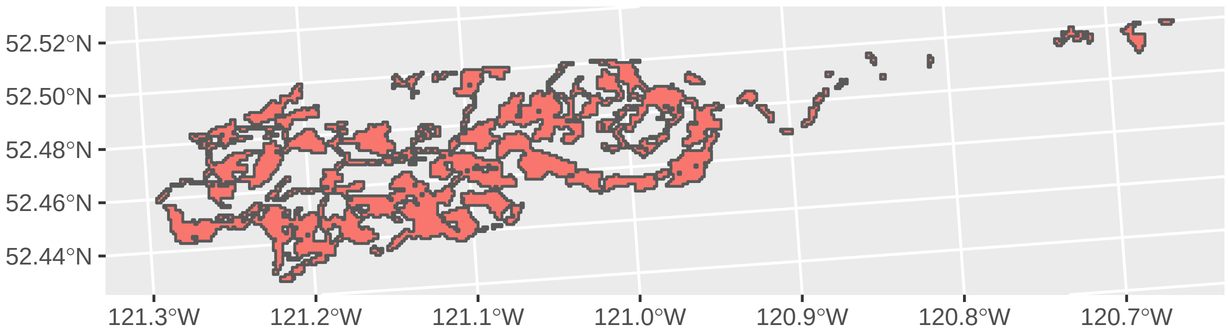
1.3. Quesnel Lake: Derive Species Raster

```
lead_htop_sv_quesnel = as.polygons(lead_htop_rast_quesnel)
lead_htop_sf_quesnel = sf::st_as_sf(lead_htop_sv_quesnel)
vri_sf = read_sf("/media/seamus/128GB_WORKD/data/vector/vri/vri_tcc_lquan_20220627.shp")
#vri_species = vri_sf[c("SPECIES_CD", "SPECIES_PC")]
vri_species =  dplyr::filter(vri_sf,
    SPECIES_CD=='PL' | SPECIES_CD=='PLI' | SPECIES_CD=='FD' | SPECIES_CD=='FDI' |
    SPECIES_CD=='SB' | SPECIES_CD=='SE' | SPECIES_CD=='SW' | SPECIES_CD=='SX' |
    SPECIES_CD=='CW' | SPECIES_CD=='HW' | SPECIES_CD=='BL' | SPECIES_CD=='LW')
vri_species_allspecies = vri_species
vri_species_allspecies$SPECIES_CD = dplyr::recode(vri_species$SPECIES_CD,
  PL = 0, PLI = 0, SB = 1, SE = 1, SW = 1, SX = 1, FD = 2, FDI = 2, CW = 3, HW = 4, BL = 5, LW = 6)
vri_species_allspecies = dplyr::rename(vri_species_allspecies, species_class = SPECIES_CD)
vri_species_allspecies = vri_species_allspecies["species_class"]
vri_species_allspecies_sf = sf::st_as_sf(vri_species_allspecies)
vri_species_aoi_quesnel = st_intersection(vri_species_allspecies_sf, st_make_valid(lead_htop_sf_quesnel
species_class_rast_quesnel = terra::rasterize(vect(vri_species_aoi_quesnel), lead_htop_rast_quesnel, fi
species_class_rast_quesnel = terra::resample(species_class_rast_quesnel, lead_htop_rast_quesnel)
species_class_raster_quesnel = raster::raster(species_class_rast_quesnel)
raster::writeRaster(species_class_raster_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/
```



1.4. Quesnel Lake: Derive Mask to Include Estimates of "RESULTS, TCC_Blocks" Areas

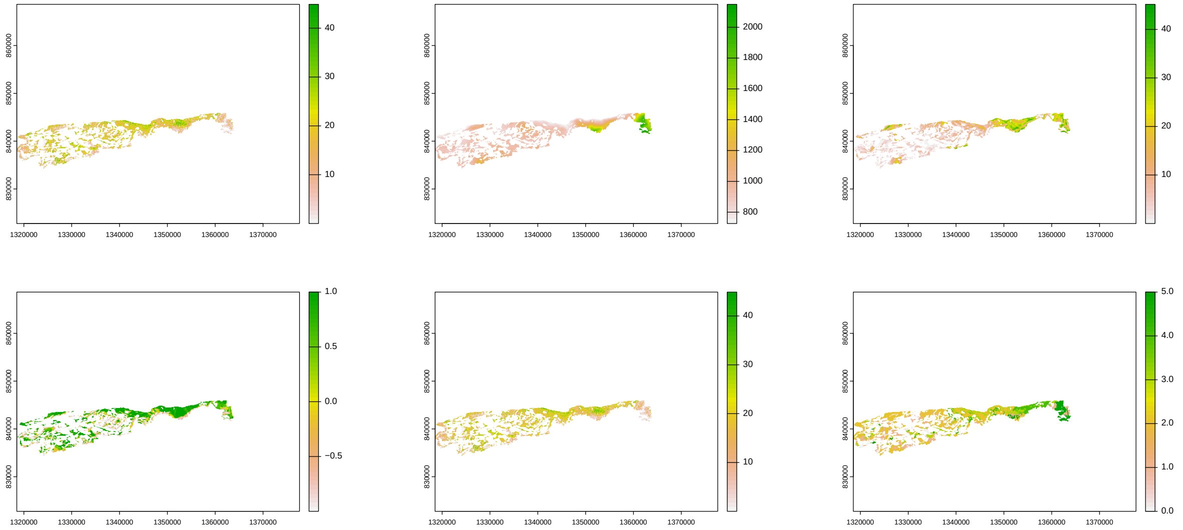
```
#mask_burn2017 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2018 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2021 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2017 = mask_burn2017["BurnSev"]
#mask_burn2018 = mask_burn2018["BurnSev"]
#mask_burn2021 = mask_burn2021["BurnSev"]
#mask_burn2017 = dplyr::filter(mask_burn2017, BurnSev == 'High')
#mask_burn2018 = dplyr::filter(mask_burn2018, BurnSev == 'High')
#mask_burn2021 = dplyr::filter(mask_burn2021, BurnSev == 'High')
#mask_roads = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Roads.shp")
#mask_roads = sf::st_zm(mask_roads)
#mask_roads = sf::st_buffer(mask_roads, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", joinStyle = "R
#mask_roads_ften = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/FTEN_Roads_All.sh
#mask_roads_ften = sf::st_zm(mask_roads_ften)
#mask_roads_ften = sf::st_buffer(mask_roads_ften, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", join
#mask_clearcut = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/RSLT_CCRES_CLEAR.sh
#mask_blocks = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Blocks_Join.shp")

#mask_burn2017_devBlocks_quesnel = sf::st_intersection(sf::st_make_valid(mask_burn2017), lead_htop_sf_q
#mask_burn2018_devBlocks_quesnel = sf::st_intersection(sf::st_make_valid(mask_burn2018), lead_htop_sf_q
#mask_burn2021_devBlocks_quesnel = sf::st_intersection(sf::st_make_valid(mask_burn2021), lead_htop_sf_q
#masks_df_devBlocks_quesnel = full_join(as_tibble(mask_burn2017_devBlocks_quesnel), as_tibble(mask_burn
#masks_sf_devBlocks_quesnel = st_as_sf(masks_df_devBlocks_quesnel)
#mask_roads_devBlocks_quesnel = sf::st_intersection(mask_roads, st_make_valid(lead_htop_sf_quesnel))
#mask_roads_ften_devBlocks_quesnel = sf::st_intersection(mask_roads_ften, st_make_valid(lead_htop_sf_qu
#masks_df_devBlocks_quesnel = full_join(as_tibble(masks_sf_devBlocks_quesnel), as_tibble(mask_roads_dev
#masks_sf_devBlocks_quesnel = st_as_sf(masks_df_devBlocks_quesnel)
masks_lquan_20220627 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/tcc_masks_lqua
masks_sf_devBlocks_quesnel = sf::st_intersection(masks_lquan_20220627, st_make_valid(lead_htop_sf_quesne
masks_rast_devBlocks_quesnel = rasterize(vect(masks_sf_devBlocks_quesnel), lead_htop_rast_quesnel, touc
masks_raster_devBlocks_quesnel = raster::raster(masks_rast_devBlocks_quesnel)
writeRaster(masks_raster_devBlocks_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs
ggplot(masks_sf_devBlocks_quesnel) + geom_sf(aes(fill = 'red'), show.legend = FALSE)
```



1.5. Quesnel Lake: Apply Masking

```
# mask by mask
lead_htop_rast_quesnel = terra::resample(lead_htop_rast_quesnel, elev_rast_quesnel)
masks_rast_devBlocks_quesnel = terra::resample(masks_rast_devBlocks_quesnel, elev_rast_quesnel)
species_class_rast_quesnel = terra::resample(species_class_rast_quesnel, elev_rast_quesnel)
lead_htop_rast_quesnel_masked = mask(lead_htop_rast_quesnel, masks_rast_devBlocks_quesnel, inverse=TRUE)
elev_rast_quesnel_masked = mask(elev_rast_quesnel, masks_rast_devBlocks_quesnel, inverse=TRUE)
slope_rast_quesnel_masked = mask(slope_rast_quesnel, masks_rast_devBlocks_quesnel, inverse=TRUE)
asp_cos_rast_quesnel_masked = mask(asp_cos_rast_quesnel, masks_rast_devBlocks_quesnel, inverse=TRUE)
asp_sin_rast_quesnel_masked = mask(asp_sin_rast_quesnel, masks_rast_devBlocks_quesnel, inverse=TRUE)
```

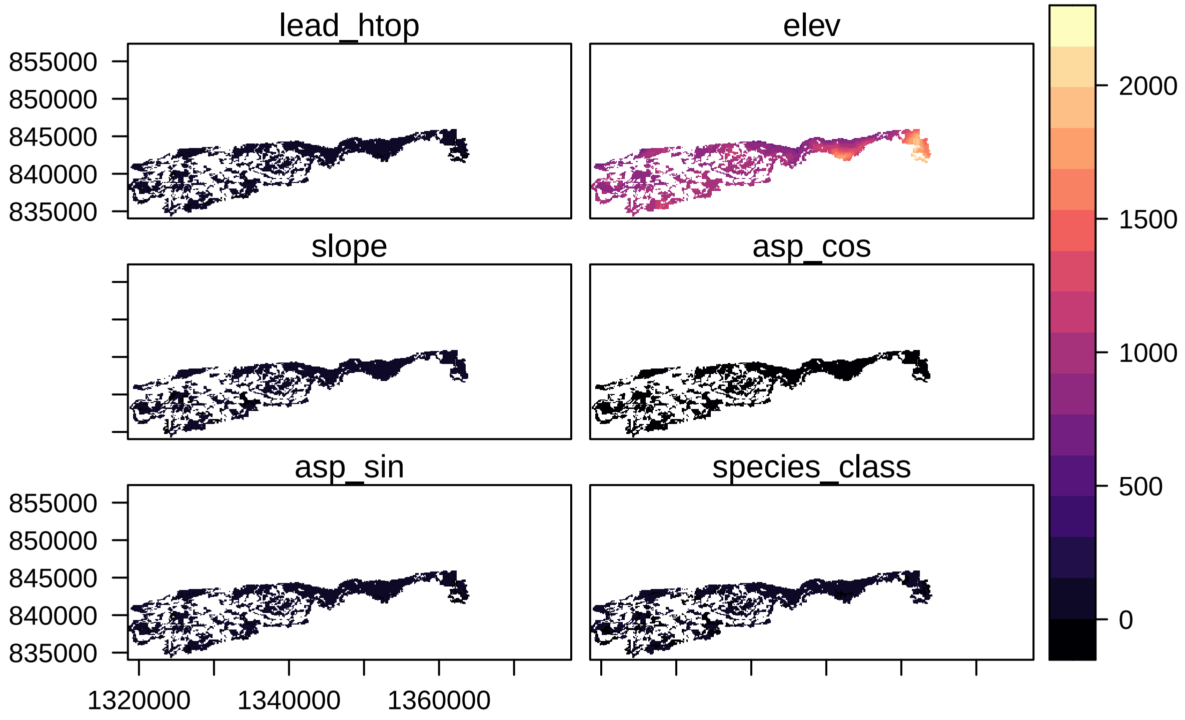
```
species_class_rast_quesnel_masked = mask(species_class_rast_quesnel, masks_rast_devBlocks_quesnel, inver
# mask by species
lead_htop_rast_quesnel_masked = mask(lead_htop_rast_quesnel_masked, species_class_rast_quesnel_masked, i
elev_rast_quesnel_masked = mask(elev_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FAL
slope_rast_quesnel_masked = mask(slope_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=F
asp_cos_rast_quesnel_masked = mask(asp_cos_rast_quesnel_masked, species_class_rast_quesnel_masked, inver
asp_sin_rast_quesnel_masked = mask(asp_sin_rast_quesnel_masked, species_class_rast_quesnel_masked, inver
# save outputs
writeRaster(lead_htop_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/
writeRaster(elev_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/maske
writeRaster(slope_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/mask
writeRaster(asp_cos_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/ma
writeRaster(asp_sin_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/ma
writeRaster(species_class_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inp
```



1.6. Quesnel Lake: Stack and Tidy Covariates

```
# Rename rasters
names(lead_htop_rast_quesnel_masked) = "lead_htop"
names(elev_rast_quesnel_masked) = "elev"
names(slope_rast_quesnel_masked) = "slope"
names(asp_cos_rast_quesnel_masked) = "asp_cos"
names(asp_sin_rast_quesnel_masked) = "asp_sin"
names(species_class_rast_quesnel_masked) = "species_class"
# transform spatRaster to raster
lead_htop_raster_quesnel_masked = raster::raster(lead_htop_rast_quesnel_masked)
elev_raster_quesnel_masked = raster::raster(elev_rast_quesnel_masked)
slope_raster_quesnel_masked = raster::raster(slope_rast_quesnel_masked)
asp_cos_raster_quesnel_masked = raster::raster(asp_cos_rast_quesnel_masked)
asp_sin_raster_quesnel_masked = raster::raster(asp_sin_rast_quesnel_masked)
species_class_raster_quesnel_masked = raster::raster(species_class_rast_quesnel_masked)
# stack rasters
```

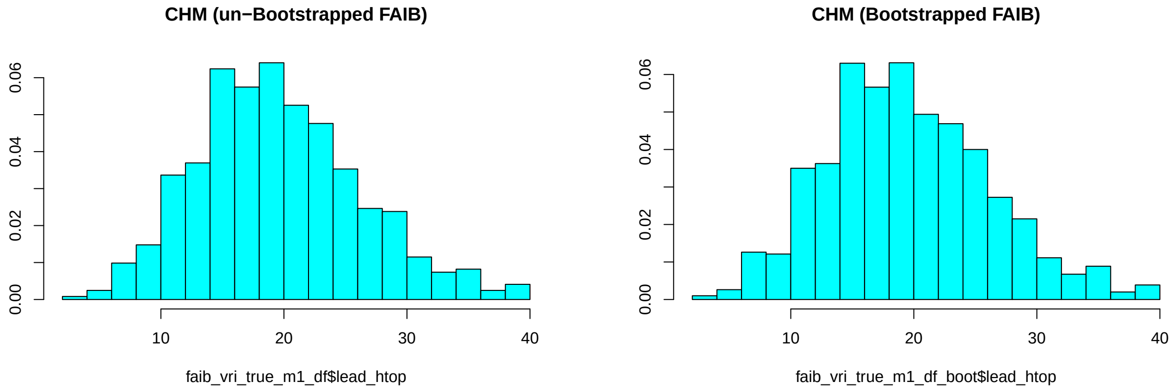
```r
covs_m1_quesnel = raster::stack(
  lead_htop_raster_quesnel_masked,
  elev_raster_quesnel_masked,
  slope_raster_quesnel_masked,
  asp_cos_raster_quesnel_masked,
  asp_sin_raster_quesnel_masked,
  species_class_raster_quesnel_masked)
# visualize
rasterVis::levelplot(covs_m1_quesnel)
```



1.7. Quesnel Lake: Bootstrapping and training/test splits

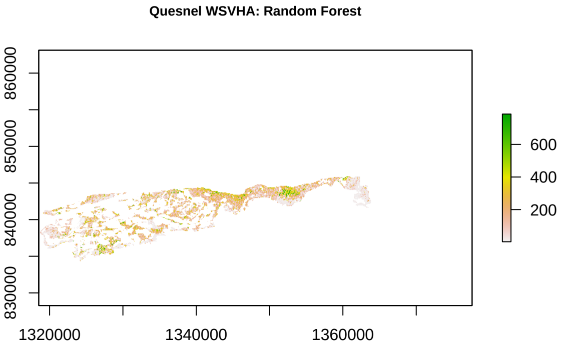
```r
# Data cleaning
faib_psp <- read.csv("/media/seamus/128GB_WORKD/EFI-TCC/0_Caret_Predict_to_writeRasterOutput/Data/FAIB_
faib_psp = subset(faib_psp, util == '12.5')
faib_psp$spc_live1 = as.factor(faib_psp$spc_live1)
faib_psp =  subset(
  faib_psp, spc_live1=='PL' | spc_live1=='PLI' | spc_live1=='FD'| spc_live1=='FDI' |
    spc_live1=='SB' | spc_live1=='SE' | spc_live1=='SW' | spc_live1=='SX' |
    spc_live1=='CW' | spc_live1=='HW' | spc_live1=='BL' | spc_live1=='LW')
faib_psp$species_class = dplyr::recode(
  faib_psp$spc_live1, PL = 1, PLI = 1, SB = 2, SE = 2, SX = 2,
  FD = 3, FDI = 3, CW = 3, HW = 4, BL = 5, LW = 6)
```

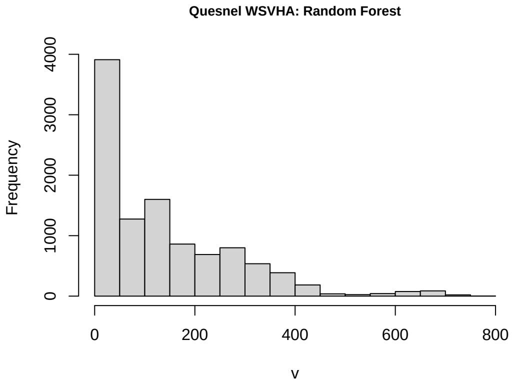
```r
faib_psp$asp_cos = cos((faib_psp$aspect * pi) / 180)
faib_psp$asp_sin = sin((faib_psp$aspect * pi) / 180)
faib_psp$elev[faib_psp$elev <= 0] = NA
faib_psp$slope[faib_psp$slope <= 0] = NA
faib_psp$lead_htop[faib_psp$lead_htop < 2] = NA
faib_psp$stemsha_L[faib_psp$stemsha_L <= 0] = NA
faib_psp$wsvha_L[faib_psp$wsvha_L <= 0] = NA
faib_psp = subset(faib_psp, stemsha_L < 864)
faib_psp$elev = as.numeric(faib_psp$elev)
faib_psp$slope = as.numeric(faib_psp$slope)
faib_psp$asp_cos = as.numeric(faib_psp$asp_cos)
faib_psp$asp_sin = as.numeric(faib_psp$asp_sin)
faib_psp$lead_htop = as.numeric(faib_psp$lead_htop)
faib_psp$species_class = as.numeric(faib_psp$species_class)
faib_psp$wsvha_L = as.numeric(faib_psp$wsvha_L)
faib_vri_true_m1_df = faib_psp[c("elev", "slope", "asp_cos", "asp_sin", "lead_htop", "species_class", "
faib_vri_true_m1_df = na.omit(faib_vri_true_m1_df)
# Bootstrapping: Weighting done by full 16-area raster payload
lead_htop_raster_all_masked = raster::raster("/media/seamus/128GB_WORKD/data/raster/tcc/inputs/masked-c
lead_htop_df_all = as.data.frame(rasterToPoints(lead_htop_raster_all_masked))
dens.fun = approxfun(density(lead_htop_df_all$lead_htop_rast_fullMask_allAreas_masked, adjust=0.8))
B = 1000
n = 4
faib_vri_true_m1_df_boot = dplyr::sample_n(faib_vri_true_m1_df, B * n, weight_by = dens.fun(faib_vri_tru
faib_vri_true_m1_df_boot = na.omit(faib_vri_true_m1_df_boot)
truehist(faib_vri_true_m1_df$lead_htop, main="CHM (un-Bootstrapped FAIB)")
truehist(faib_vri_true_m1_df_boot$lead_htop, main="CHM (Bootstrapped FAIB)")

faib_vri_true_m1_df_boot_split = createDataPartition(faib_vri_true_m1_df_boot$wsvha_L, p=0.80, list=F)
faib_vri_true_m1_df_split = createDataPartition(faib_vri_true_m1_df$wsvha_L, p=0.80, list=F)

train_m1_boot = faib_vri_true_m1_df_boot[faib_vri_true_m1_df_boot_split, ]
test_m1_boot = faib_vri_true_m1_df_boot[-faib_vri_true_m1_df_boot_split, ]
train_m1 = faib_vri_true_m1_df[faib_vri_true_m1_df_split, ]
test_m1 = faib_vri_true_m1_df[-faib_vri_true_m1_df_split, ]

X_train_m1_boot = train_m1_boot[,-7]
y_train_m1_boot = train_m1_boot[, 7]
X_test_m1_boot = test_m1_boot[,-7]
y_test_m1_boot = test_m1_boot[, 7]
X_m1_boot = faib_vri_true_m1_df_boot[,-7]
y_m1_boot = faib_vri_true_m1_df_boot[, 7]

X_train_m1 = train_m1[,-7]
y_train_m1 = train_m1[, 7]
X_test_m1 = test_m1[,-7]
y_test_m1 = test_m1[, 7]
X_m1 = faib_vri_true_m1_df[,-7]
y_m1 = faib_vri_true_m1_df[, 7]
```



1.8. Quesnel Lake: Modeling WSVHA Estimates with Random Forest Regression

```
tuneResult_rf_m1_full <- tune.randomForest(
  X_m1_boot, y_m1_boot,
  mtry = c(2:10), ntree = 50,
  tunecontrol = tune.control(sampling = "cross", cross = 10),
  preProcess = c('YeoJohnson', 'scale', 'center', 'corr'))

tuneResult_rf_m1_train <- tune.randomForest(
  X_train_m1_boot, y_train_m1_boot,
  mtry = c(2:10), ntree = 50,
  tunecontrol = tune.control(sampling = "cross", cross = 10),
  preProcess = c('YeoJohnson', 'scale', 'center', 'corr'))
tunedModel_rf_m1_full <- tuneResult_rf_m1_full$best.model
tunedModel_rf_m1_train <- tuneResult_rf_m1_train$best.model
tunedModel_rf_m1 = predict(tunedModel_rf_m1_full, newdata=faib_vri_true_m1_df, type = "response")
tunedModel_rf_m1_test = predict(tunedModel_rf_m1_train, newdata=test_m1, type = "response")
save(tunedModel_rf_m1_full, file = "/media/seamus/128GB_WORKD/data/models/tcc-wsvha/wsvha_model1_random
tuneResult_rf_m1_full
R2(tunedModel_rf_m1, faib_vri_true_m1_df$wsvha_L)
MAE(tunedModel_rf_m1, faib_vri_true_m1_df$wsvha_L)
RMSE(tunedModel_rf_m1, faib_vri_true_m1_df$wsvha_L)
MAE(tunedModel_rf_m1_test, test_m1$wsvha_L)
RMSE(tunedModel_rf_m1_test, test_m1$wsvha_L)
wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks <- raster::predict(covs_m1_que
wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks$layer[wsvha_model1_randomFores
writeRaster(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, overwrite=TRUE,
  filename = "/media/seamus/128GB_WORKD/data/raster/tcc/outputs/wsvha/bootstrapped/wsvha_model1_randomF
plot(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, main="Quesnel WSVHA: Ran
hist(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, main="Quesnel WSVHA: Ran
```





2. All-Species Areas: Includes Fd and non-Fd stands

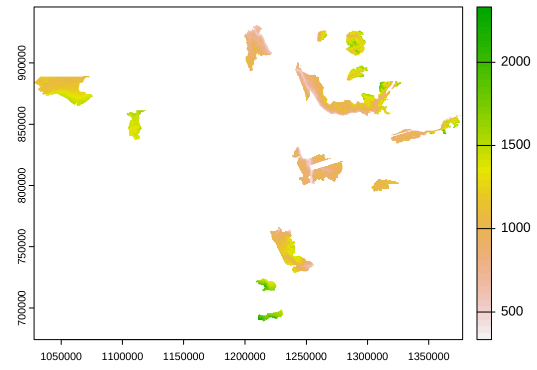
2.1. All-Species Areas: Import DEM and CHM

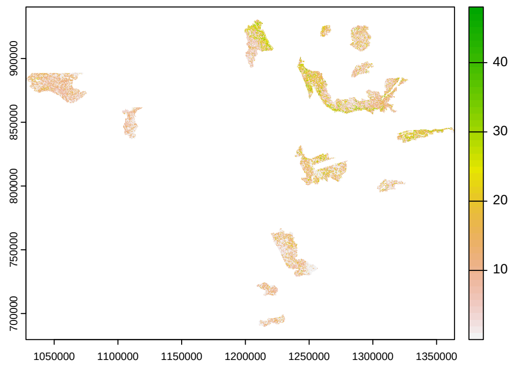
```
# Merge chunks
filez_be_ahbau = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/ahbau/BareEarth", full.names =
elev_raster_list_ahbau <- lapply(filez_be_ahbau, raster)
elev_raster_ahbau = do.call(merge, c(elev_raster_list_ahbau, tolerance = 1))
elev_rast_ahbau = terra::rast(elev_raster_ahbau)
terra::crs(elev_rast_ahbau) = "epsg:3005"
elev_rast_ahbau = terra::aggregate(elev_rast_ahbau, fact = 100, fun = mean)
writeRaster(elev_rast_ahbau, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/ahbau/elev_raster

filez_be_bells = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/bells/BareEarth", full.names =
elev_raster_list_bells <- lapply(filez_be_bells, raster)
elev_raster_bells = do.call(merge, c(elev_raster_list_bells, tolerance = 1))
elev_rast_bells = terra::rast(elev_raster_bells)
terra::crs(elev_rast_bells) = "epsg:3005"
elev_rast_bells = terra::aggregate(elev_rast_bells, fact = 100, fun = mean)
writeRaster(elev_rast_bells, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/bells/elev_raster

filez_be_big_valley = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_valley/BareEarth", f
elev_raster_list_big_valley <- lapply(filez_be_big_valley, raster)
elev_raster_big_valley = do.call(merge, c(elev_raster_list_big_valley, tolerance = 1))
elev_rast_big_valley = terra::rast(elev_raster_big_valley)
terra::crs(elev_rast_big_valley) = "epsg:3005"
elev_rast_big_valley = terra::aggregate(elev_rast_big_valley, fact = 100, fun = mean)
writeRaster(elev_rast_big_valley, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_valley/e

filez_be_cariboo_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/cariboo_lake/BareEarth
elev_raster_list_cariboo_lake <- lapply(filez_be_cariboo_lake, raster)
elev_raster_cariboo_lake = do.call(merge, c(elev_raster_list_cariboo_lake, tolerance = 1))
elev_rast_cariboo_lake = terra::rast(elev_raster_cariboo_lake)
terra::crs(elev_rast_cariboo_lake) = "epsg:3005"
elev_rast_cariboo_lake = terra::aggregate(elev_rast_cariboo_lake, fact = 100, fun = mean)
writeRaster(elev_rast_cariboo_lake, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/cariboo_la
filez_be_charleson_marvincreek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/charleson_marv
elev_raster_list_charleson_marvincreek <- lapply(filez_be_charleson_marvincreek, raster)
elev_raster_charleson_marvincreek = do.call(merge, c(elev_raster_list_charleson_marvincreek, tolerance =
elev_rast_charleson_marvincreek = terra::rast(elev_raster_charleson_marvincreek)
terra::crs(elev_rast_charleson_marvincreek) = "epsg:3005"
elev_rast_charleson_marvincreek = terra::aggregate(elev_rast_charleson_marvincreek, fact = 100, fun = me
writeRaster(elev_rast_charleson_marvincreek, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/ch

filez_be_dash = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/dash/BareEarth", full.names = T
elev_raster_list_dash <- lapply(filez_be_dash, raster)
elev_raster_dash = do.call(merge, c(elev_raster_list_dash, tolerance = 1))
elev_rast_dash = terra::rast(elev_raster_dash)
terra::crs(elev_rast_dash) = "epsg:3005"
elev_rast_dash = terra::aggregate(elev_rast_dash, fact = 100, fun = mean)
writeRaster(elev_rast_dash, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/dash/elev_raster_10

filez_be_little_river = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_river/BareEarth
elev_raster_list_little_river <- lapply(filez_be_little_river, raster)
elev_raster_little_river = do.call(merge, c(elev_raster_list_little_river, tolerance = 1))
elev_rast_little_river = terra::rast(elev_raster_little_river)
terra::crs(elev_rast_little_river) = "epsg:3005"
elev_rast_little_river = terra::aggregate(elev_rast_little_river, fact = 100, fun = mean)
writeRaster(elev_rast_little_river, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_rive

filez_be_little_swift = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_swift/BareEarth
elev_raster_list_little_swift <- lapply(filez_be_little_swift, raster)
elev_raster_little_swift = do.call(merge, c(elev_raster_list_little_swift, tolerance = 1))
elev_rast_little_swift = terra::rast(elev_raster_little_swift)
terra::crs(elev_rast_little_swift) = "epsg:3005"
elev_rast_little_swift = terra::aggregate(elev_rast_little_swift, fact = 100, fun = mean)
writeRaster(elev_rast_little_swift, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_swif

filez_be_mcintosh = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/mcintosh/BareEarth", full.n
elev_raster_list_mcintosh <- lapply(filez_be_mcintosh, raster)
elev_raster_mcintosh = do.call(merge, c(elev_raster_list_mcintosh, tolerance = 1))
elev_rast_mcintosh = terra::rast(elev_raster_mcintosh)
terra::crs(elev_rast_mcintosh) = "epsg:3005"
elev_rast_mcintosh = terra::aggregate(elev_rast_mcintosh, fact = 100, fun = mean)
writeRaster(elev_rast_mcintosh, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/mcintosh/elev_r

filez_be_phillips_anahim_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phillips_anahim
elev_raster_phillips_anahim_lake = do.call(merge, c(elev_raster_list_phillips_anahim_lake, tolerance = 1
elev_rast_phillips_anahim_lake = terra::rast(elev_raster_phillips_anahim_lake)
terra::crs(elev_rast_phillips_anahim_lake) = "epsg:3005"
elev_rast_phillips_anahim_lake = terra::aggregate(elev_rast_phillips_anahim_lake, fact = 100, fun = mean
writeRaster(elev_rast_phillips_anahim_lake, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phi

filez_be_piltz = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/piltz/BareEarth", full.names =
elev_raster_list_piltz = lapply(filez_be_piltz, raster)
elev_raster_piltz = do.call(merge, c(elev_raster_list_piltz, tolerance = 1))
elev_rast_piltz = terra::rast(elev_raster_piltz)
terra::crs(elev_rast_piltz) = "epsg:3005"
elev_rast_piltz = terra::aggregate(elev_rast_piltz, fact = 100, fun = mean)
```

```
writeRaster(elev_rast_piltz, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/piltz/elev_raster_

filez_be_punky_clisbako = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punky_clisbako/BareEa
elev_raster_punky_clisbako = do.call(merge, c(elev_raster_list_punky_clisbako, tolerance = 1))
elev_rast_punky_clisbako = terra::rast(elev_raster_punky_clisbako)
terra::crs(elev_rast_punky_clisbako) = "epsg:3005"
elev_rast_punky_clisbako = terra::aggregate(elev_rast_punky_clisbako, fact = 100, fun = mean)
writeRaster(elev_rast_punky_clisbako, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punky_cli

filez_vh_ahbau = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/ahbau/VegHt", full.names = T,
lead_htop_raster_list_ahbau <- lapply(filez_vh_ahbau, raster)
lead_htop_raster_ahbau = do.call(merge, c(lead_htop_raster_list_ahbau, tolerance = 1))
lead_htop_rast_ahbau = terra::rast(lead_htop_raster_ahbau)
terra::crs(lead_htop_rast_ahbau) = "epsg:3005"
lead_htop_rast_ahbau = terra::aggregate(lead_htop_rast_ahbau, fact = 100, fun = mean)
writeRaster(lead_htop_rast_ahbau, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/ahbau/lead_ht

filez_vh_bells = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/bells/VegHt", full.names = T,
lead_htop_raster_list_bells <- lapply(filez_vh_bells, raster)
lead_htop_raster_bells = do.call(merge, c(lead_htop_raster_list_bells, tolerance = 1))
lead_htop_rast_bells = terra::rast(lead_htop_raster_bells)
terra::crs(lead_htop_rast_bells) = "epsg:3005"
lead_htop_rast_bells = terra::aggregate(lead_htop_rast_bells, fact = 100, fun = mean)
writeRaster(lead_htop_rast_bells, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/bells/lead_ht

filez_vh_big_valley = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_valley/VegHt", full.n
lead_htop_raster_list_big_valley <- lapply(filez_vh_big_valley, raster)
lead_htop_raster_big_valley = do.call(merge, c(lead_htop_raster_list_big_valley, tolerance = 1))
lead_htop_rast_big_valley = terra::rast(lead_htop_raster_big_valley)
terra::crs(lead_htop_rast_big_valley) = "epsg:3005"
lead_htop_rast_big_valley = terra::aggregate(lead_htop_rast_big_valley, fact = 100, fun = mean)
writeRaster(lead_htop_rast_big_valley, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_vall

filez_vh_cariboo_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/cariboo_lake/VegHt", fu
lead_htop_raster_list_cariboo_lake <- lapply(filez_vh_cariboo_lake, raster)
lead_htop_raster_cariboo_lake = do.call(merge, c(lead_htop_raster_list_cariboo_lake, tolerance = 1))
lead_htop_rast_cariboo_lake = terra::rast(lead_htop_raster_cariboo_lake)
terra::crs(lead_htop_rast_cariboo_lake) = "epsg:3005"
lead_htop_rast_cariboo_lake = terra::aggregate(lead_htop_rast_cariboo_lake, fact = 100, fun = mean)
writeRaster(lead_htop_rast_cariboo_lake, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/caribo

filez_vh_charleson_marvincreek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/charleson_marv
lead_htop_raster_list_charleson_marvincreek <- lapply(filez_vh_charleson_marvincreek, raster)
lead_htop_raster_charleson_marvincreek = do.call(merge, c(lead_htop_raster_list_charleson_marvincreek, t
lead_htop_rast_charleson_marvincreek = terra::rast(lead_htop_raster_charleson_marvincreek)
terra::crs(lead_htop_rast_charleson_marvincreek) = "epsg:3005"
lead_htop_rast_charleson_marvincreek = terra::aggregate(lead_htop_rast_charleson_marvincreek, fact = 100
writeRaster(lead_htop_rast_charleson_marvincreek, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosai

filez_vh_dash = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/dash/VegHt", full.names = T, al
lead_htop_raster_list_dash <- lapply(filez_vh_dash, raster)
lead_htop_raster_dash = do.call(merge, c(lead_htop_raster_list_dash, tolerance = 1))
lead_htop_rast_dash = terra::rast(lead_htop_raster_dash)
```

```
terra::crs(lead_htop_rast_dash) = "epsg:3005"
lead_htop_rast_dash = terra::aggregate(lead_htop_rast_dash, fact = 100, fun = mean)
writeRaster(lead_htop_rast_dash, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/le

filez_vh_little_river = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_river/VegHt", fu
lead_htop_raster_list_little_river <- lapply(filez_vh_little_river, raster)
lead_htop_raster_little_river = do.call(merge, c(lead_htop_raster_list_little_river, tolerance = 1))
lead_htop_rast_little_river = terra::rast(lead_htop_raster_little_river)
terra::crs(lead_htop_rast_little_river) = "epsg:3005"
lead_htop_rast_little_river = terra::aggregate(lead_htop_rast_little_river, fact = 100, fun = mean)
writeRaster(lead_htop_rast_little_river, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/littl

filez_vh_little_swift = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_swift/VegHt", fu
lead_htop_raster_list_little_swift <- lapply(filez_vh_little_swift, raster)
lead_htop_raster_little_swift = do.call(merge, c(lead_htop_raster_list_little_swift, tolerance = 1))
lead_htop_rast_little_swift = terra::rast(lead_htop_raster_little_swift)
terra::crs(lead_htop_rast_little_swift) = "epsg:3005"
lead_htop_rast_little_swift = terra::aggregate(lead_htop_rast_little_swift, fact = 100, fun = mean)
writeRaster(lead_htop_rast_little_swift, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/littl

filez_vh_mcintosh = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/mcintosh/VegHt", full.names
lead_htop_raster_list_mcintosh <- lapply(filez_vh_mcintosh, raster)
lead_htop_raster_mcintosh = do.call(merge, c(lead_htop_raster_list_mcintosh, tolerance = 1))
lead_htop_rast_mcintosh = terra::rast(lead_htop_raster_mcintosh)
terra::crs(lead_htop_rast_mcintosh) = "epsg:3005"
lead_htop_rast_mcintosh = terra::aggregate(lead_htop_rast_mcintosh, fact = 100, fun = mean)
writeRaster(lead_htop_rast_mcintosh, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/mcintosh/l

filez_vh_phillips_anahim_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phillips_anahim
lead_htop_raster_list_phillips_anahim_lake <- lapply(filez_vh_phillips_anahim_lake, raster)
lead_htop_raster_phillips_anahim_lake = do.call(merge, c(lead_htop_raster_list_phillips_anahim_lake, tol
lead_htop_rast_phillips_anahim_lake = terra::rast(lead_htop_raster_phillips_anahim_lake)
terra::crs(lead_htop_rast_phillips_anahim_lake) = "epsg:3005"
lead_htop_rast_phillips_anahim_lake = terra::aggregate(lead_htop_rast_phillips_anahim_lake, fact = 100,
writeRaster(lead_htop_rast_phillips_anahim_lake, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaic

filez_vh_piltz = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/piltz/VegHt", full.names = T,
lead_htop_raster_list_piltz <- lapply(filez_vh_piltz, raster)
lead_htop_raster_piltz = do.call(merge, c(lead_htop_raster_list_piltz, tolerance = 1))
lead_htop_rast_piltz = terra::rast(lead_htop_raster_piltz)
terra::crs(lead_htop_rast_piltz) = "epsg:3005"extent_all_sf
lead_htop_rast_piltz = terra::aggregate(lead_htop_rast_piltz, fact = 100, fun = mean)
writeRaster(lead_htop_rast_piltz, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/piltz/lead_ht

filez_vh_punky_clisbako = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punky_clisbako/VegHt
lead_htop_raster_list_punky_clisbako <- lapply(filez_vh_punky_clisbako, raster)
lead_htop_raster_punky_clisbako = do.call(merge, c(lead_htop_raster_list_punky_clisbako, tolerance = 1))
lead_htop_rast_punky_clisbako = terra::rast(lead_htop_raster_punky_clisbako)
terra::crs(lead_htop_rast_punky_clisbako) = "epsg:3005"
lead_htop_rast_punky_clisbako = terra::aggregate(lead_htop_rast_punky_clisbako, fact = 100, fun = mean)
writeRaster(lead_htop_rast_punky_clisbako, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punk

# Merge mosaics
```

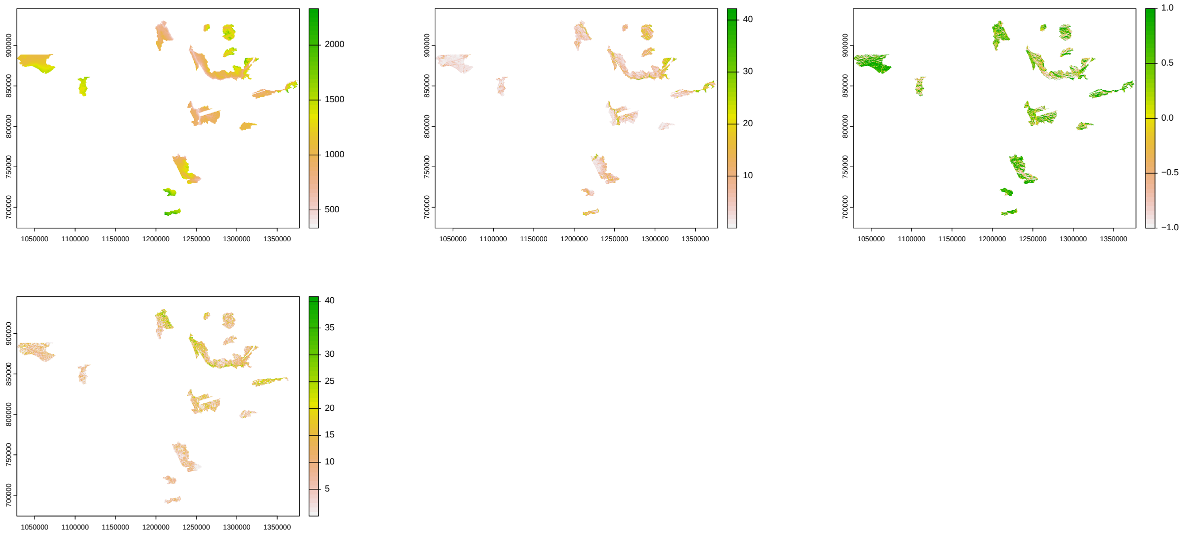
```
elev_raster_allSpeciesAreas_list = list(
  elev_raster_ahbau, #elev_raster_bells,
  elev_raster_big_valley, elev_raster_cariboo_lake,
  elev_raster_charleson_marvincreek, elev_raster_dash,
  elev_raster_hawks_creek, elev_raster_little_river,
  elev_raster_little_swift, elev_raster_mcintosh,
  elev_raster_phillips_anahim_lake, elev_raster_piltz,
  elev_raster_punky_clisbako)

elev_raster_allSpeciesAreas = do.call(merge, c(elev_raster_allSpeciesAreas_list, tolerance = 1))
elev_rast_allSpeciesAreas = terra::rast(elev_raster_allSpeciesAreas)
writeRaster(elev_raster_allSpeciesAreas, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/elev_

lead_htop_raster_allSpeciesAreas_list = list(
  lead_htop_raster_ahbau, #lead_htop_raster_bells,
  lead_htop_raster_big_valley, lead_htop_raster_cariboo_lake,
  lead_htop_raster_charleson_marvincreek, lead_htop_raster_dash,
  lead_htop_raster_little_river, lead_htop_raster_little_swift,
  lead_htop_raster_mcintosh, lead_htop_raster_phillips_anahim_lake,
  lead_htop_raster_piltz, lead_htop_raster_punky_clisbako)

lead_htop_raster_allSpeciesAreas = do.call(merge, c(lead_htop_raster_allSpeciesAreas_list, tolerance = 
lead_htop_rast_allSpeciesAreas = terra::rast(lead_htop_raster_allSpeciesAreas)
writeRaster(lead_htop_raster_allSpeciesAreas, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/
```





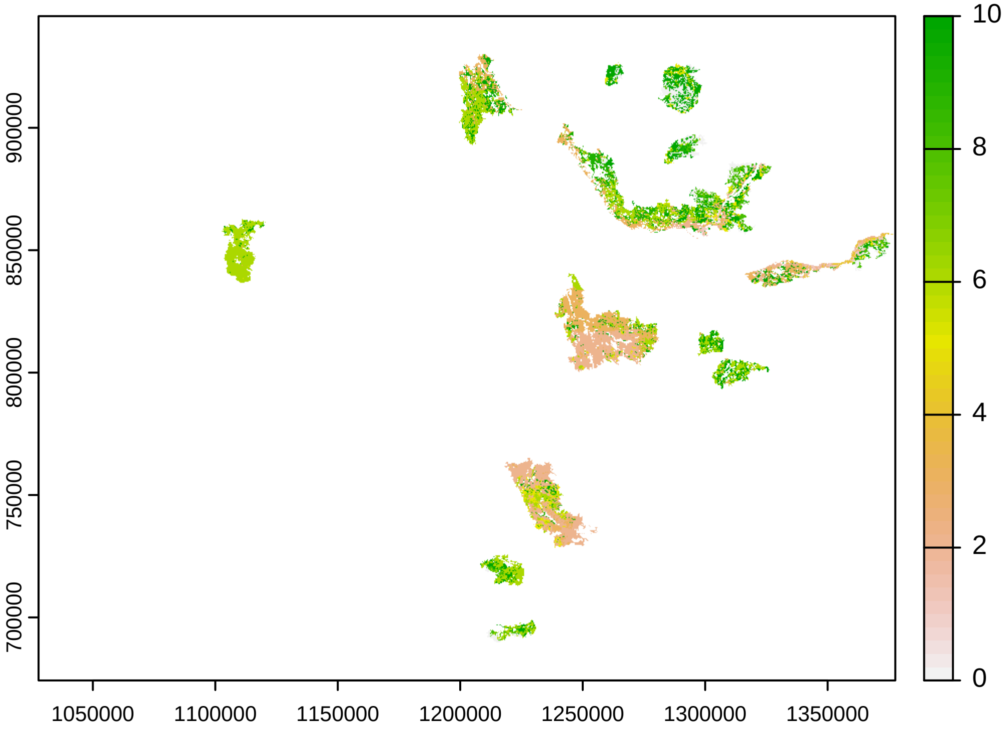
2.2. All-Species Areas: Derive Terrain Rasters

```
slope_rast_allSpeciesAreas = terra::terrain(elev_rast_allSpeciesAreas, v="slope", unit="degrees", neigh
aspect_rast_allSpeciesAreas = terra::terrain(elev_rast_allSpeciesAreas, v="aspect", unit="degrees", nei
asp_cos_rast_allSpeciesAreas = cos((aspect_rast_allSpeciesAreas*pi)/180)
asp_sin_rast_allSpeciesAreas = sin((aspect_rast_allSpeciesAreas*pi)/180)
lead_htop_rast_allSpeciesAreas = terra::resample(lead_htop_rast_allSpeciesAreas, elev_rast_allSpeciesAr
writeRaster(elev_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm
writeRaster(slope_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/un
writeRaster(asp_cos_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/
writeRaster(asp_sin_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/
writeRaster(lead_htop_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/input
```



2.3. All-Species Areas: Derive Species Raster

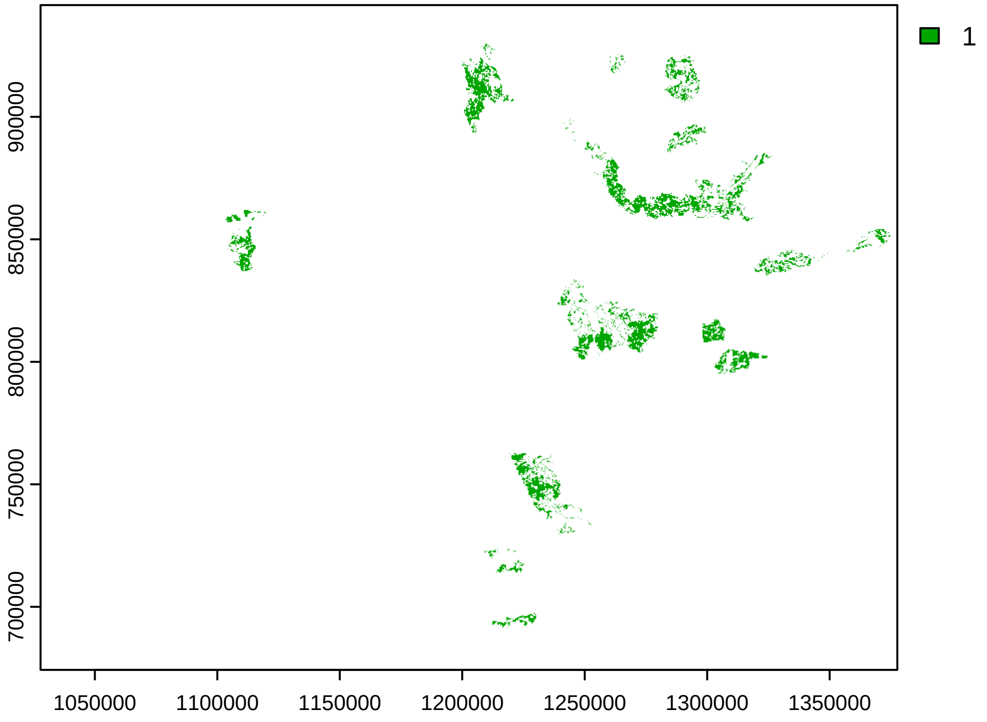
```
lead_htop_sv_allSpeciesAreas = as.polygons(lead_htop_rast_allSpeciesAreas)
lead_htop_sf_allSpeciesAreas = sf::st_as_sf(lead_htop_sv_allSpeciesAreas)
#vri_sf_allSpeciesAreas = read_sf("/media/seamus/128GB_WORKD/data/vector/vri/vri_bc_2020_rank1.shp")
vri_sf_allSpeciesAreas = read_sf("/media/seamus/128GB_WORKD/data/vector/vri/vri_tcc_lquan_20220627.shp"
vri_species_allSpeciesAreas = vri_sf_allSpeciesAreas[c("SPECIES_CD", "SPECIES_PC")]
vri_species_aoi_allSpeciesAreas =  dplyr::filter(vri_species_allSpeciesAreas,
    SPECIES_CD=='PL' | SPECIES_CD=='PLI' | SPECIES_CD=='FD' | SPECIES_CD=='FDI' |
    SPECIES_CD=='SB' | SPECIES_CD=='SE' | SPECIES_CD=='SW' | SPECIES_CD=='SX' |
    SPECIES_CD=='CW' | SPECIES_CD=='HW' | SPECIES_CD=='BL' | SPECIES_CD=='LW')
vri_species_allSpeciesAreas = vri_species_aoi_allSpeciesAreas
vri_species_allSpeciesAreas$SPECIES_CD = dplyr::recode(vri_species_allSpeciesAreas$SPECIES_CD,
  PL = 0, PLI = 0, SB = 1, SE = 1, SW = 1, SX = 1, FD = 2, FDI = 2, CW = 3, HW = 4, BL = 5, LW = 6)
vri_species_allSpeciesAreas = dplyr::rename(vri_species_allSpeciesAreas, species_class = SPECIES_CD)
vri_species_allSpeciesAreas = vri_species_allSpeciesAreas["species_class"]
vri_species_allSpeciesAreas_sf = sf::st_as_sf(vri_species_allSpeciesAreas)
plot(st_geometry(vri_species_aoi_allSpeciesAreas))
vri_species_aoi_allSpeciesAreas = st_intersection(vri_species_allSpeciesAreas_sf, st_make_valid(lead_ht
species_class_rast_allSpeciesAreas = terra::rasterize(vect(vri_species_aoi_allSpeciesAreas), lead_htop_
species_class_rast_allSpeciesAreas = terra::resample(species_class_rast_allSpeciesAreas, lead_htop_rast
species_class_raster_allSpeciesAreas = raster::raster(species_class_rast_allSpeciesAreas)
raster::writeRaster(species_class_raster_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/ra
```



2.4. All-Species Areas: Derive Mask From Approved Layers & Extents

```
#mask_burn2017 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2018 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2021 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity T
#mask_burn2017 = mask_burn2017["BurnSev"]
#mask_burn2018 = mask_burn2018["BurnSev"]
#mask_burn2021 = mask_burn2021["BurnSev"]
#mask_burn2017 = dplyr::filter(mask_burn2017, BurnSev == 'High')
#mask_burn2018 = dplyr::filter(mask_burn2018, BurnSev == 'High')
#mask_burn2021 = dplyr::filter(mask_burn2021, BurnSev == 'High')
#mask_roads = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Roads.shp")
#mask_roads = sf::st_zm(mask_roads)
#mask_roads = sf::st_buffer(mask_roads, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", joinStyle = "R
#mask_roads_ften = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/FTEN_Roads_All.sh
#mask_roads_ften = sf::st_zm(mask_roads_ften)
#mask_roads_ften = sf::st_buffer(mask_roads_ften, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", join
#mask_clearcut = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/RSLT_CCRES_CLEAR.sh
#mask_blocks = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Blocks_Join.shp")

#extent_sf = read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/Lidar_Extent_Cleaned.shp")
#extent_sfz = st_zm(extent_sf) # drop z dimensions for plotting purposes
#extent_sfz_3005 = sf::st_transform(extent_sfz, 3005)
#ggplot(extent_sfz) + geom_sf(aes(fill = 'red'), show.legend = FALSE)
```

```
#mask_burn2017_fullMask_allAreas = sf::st_intersection(sf::st_make_valid(mask_burn2017), extent_sfz_300
#mask_burn2018_fullMask_allAreas = sf::st_intersection(sf::st_make_valid(mask_burn2018), extent_sfz_300
#mask_burn2021_fullMask_allAreas = sf::st_intersection(sf::st_make_valid(mask_burn2021), extent_sfz_300
#masks_df_fullMask_allAreas = full_join(as_tibble(mask_burn2017_fullMask_allAreas), as_tibble(mask_burn
#masks_sf_fullMask_allAreas = st_as_sf(masks_df_fullMask_allAreas)
#mask_clearcut_fullMask_allAreas = sf::st_intersection(mask_clearcut, st_make_valid(extent_sfz_3005)
#masks_df_fullMask_allAreas = full_join(as_tibble(masks_sf_fullMask_allAreas), as_tibble(mask_clearcut_
#masks_sf_fullMask_allAreas = st_as_sf(masks_df_fullMask_allAreas)
#mask_blocks_fullMask_allAreas = sf::st_intersection(mask_blocks, st_make_valid(extent_sfz_3005))
#masks_df_fullMask_allAreas = full_join(as_tibble(masks_sf_fullMask_allAreas), as_tibble(mask_blocks_fu
#masks_sf_fullMask_allAreas = st_as_sf(masks_df_fullMask_allAreas)
#mask_roads_fullMask_allAreas = sf::st_intersection(mask_roads, st_make_valid(extent_sfz_3005))
#mask_roads_ften_fullMask_allAreas = sf::st_intersection(mask_roads_ften, st_make_valid(extent_sfz_3005
#masks_df_fullMask_allAreas = full_join(as_tibble(masks_sf_fullMask_allAreas), as_tibble(mask_roads_ful
#masks_sf_fullMask_allAreas = st_as_sf(masks_df_fullMask_allAreas)
masks_lquan_20220627 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/tcc_masks_lqua
#masks_sf_fullMask_allAreas = sf::st_intersection(masks_lquan_20220627, st_make_valid(lead_htop_sf_allS
masks_rast_fullMask_allAreas = rasterize(vect(masks_lquan_20220627), lead_htop_rast_allSpeciesAreas, to
masks_raster_fullMask_allAreas = raster::raster(masks_rast_fullMask_allAreas)
writeRaster(masks_raster_fullMask_allAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/input
#ggplot(masks_sf_fullMask_allAreas) + geom_sf(aes(fill = 'red'), show.legend = FALSE)
```



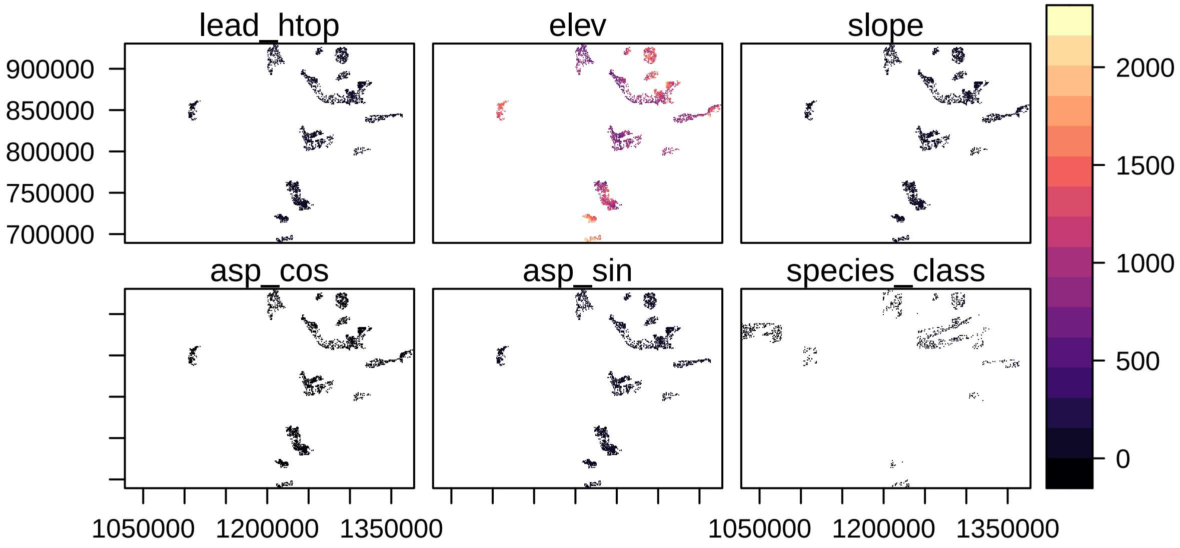
2.5. All-Species Areas: Apply Masking

```
# align
lead_htop_rast_allSpeciesAreas = terra::resample(lead_htop_rast_allSpeciesAreas, elev_rast_allSpeciesAre
masks_rast_fullMask_allAreas = terra::resample(masks_rast_fullMask_allAreas, elev_rast_allSpeciesAreas)
species_class_rast_allSpeciesAreas = terra::resample(species_class_rast_allSpeciesAreas, elev_rast_allSp
# mask by mask
lead_htop_rast_allSpeciesAreas_masked = mask(lead_htop_rast_allSpeciesAreas, masks_rast_fullMask_allArea
elev_rast_allSpeciesAreas_masked = mask(elev_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, invers
slope_rast_allSpeciesAreas_masked = mask(slope_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inve
asp_cos_rast_allSpeciesAreas_masked = mask(asp_cos_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, 
asp_sin_rast_allSpeciesAreas_masked = mask(asp_sin_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, 
species_class_rast_allSpeciesAreas_masked = mask(species_class_rast_allSpeciesAreas, masks_rast_fullMas
# mask by species
lead_htop_rast_allSpeciesAreas_masked = mask(lead_htop_rast_allSpeciesAreas_masked, species_class_rast_a
elev_rast_allSpeciesAreas_masked = mask(elev_rast_allSpeciesAreas_masked, species_class_rast_allSpecies
slope_rast_allSpeciesAreas_masked = mask(slope_rast_allSpeciesAreas_masked, species_class_rast_allSpeci
asp_cos_rast_allSpeciesAreas_masked = mask(asp_cos_rast_allSpeciesAreas_masked, species_class_rast_allSp
asp_sin_rast_allSpeciesAreas_masked = mask(asp_sin_rast_allSpeciesAreas_masked, species_class_rast_allSp
species_class_raster_allSpeciesAreas_masked = raster::raster(species_class_rast_allSpeciesAreas_masked)
writeRaster(lead_htop_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc
writeRaster(elev_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inpu
writeRaster(slope_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inp
writeRaster(asp_cos_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/
writeRaster(asp_sin_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/
writeRaster(species_class_raster_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/rast
```

2.6. All-Species Areas: Stack and Tidy Covariates

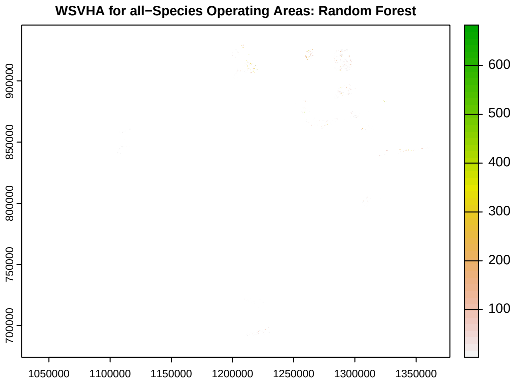
```
# Rename rasters
names(lead_htop_rast_allSpeciesAreas_masked) = "lead_htop"
names(elev_rast_allSpeciesAreas_masked) = "elev"
names(slope_rast_allSpeciesAreas_masked) = "slope"
names(asp_cos_rast_allSpeciesAreas_masked) = "asp_cos"
names(asp_sin_rast_allSpeciesAreas_masked) = "asp_sin"
names(species_class_rast_allSpeciesAreas_masked) = "species_class"
# transform spatRaster to raster
lead_htop_raster_allSpeciesAreas_masked = raster::raster(lead_htop_rast_allSpeciesAreas_masked)
elev_raster_allSpeciesAreas_masked = raster::raster(elev_rast_allSpeciesAreas_masked)
slope_raster_allSpeciesAreas_masked = raster::raster(slope_rast_allSpeciesAreas_masked)
asp_cos_raster_allSpeciesAreas_masked = raster::raster(asp_cos_rast_allSpeciesAreas_masked)
asp_sin_raster_allSpeciesAreas_masked = raster::raster(asp_sin_rast_allSpeciesAreas_masked)
species_class_raster_allSpeciesAreas_masked = raster::raster(species_class_rast_allSpeciesAreas_masked)
# stack rasters
covs_m1_allSpeciesAreas = raster::stack(
  lead_htop_raster_allSpeciesAreas_masked,
  elev_raster_allSpeciesAreas_masked,
  slope_raster_allSpeciesAreas_masked,
  asp_cos_raster_allSpeciesAreas_masked,
  asp_sin_raster_allSpeciesAreas_masked,
  species_class_raster_allSpeciesAreas_masked)
```

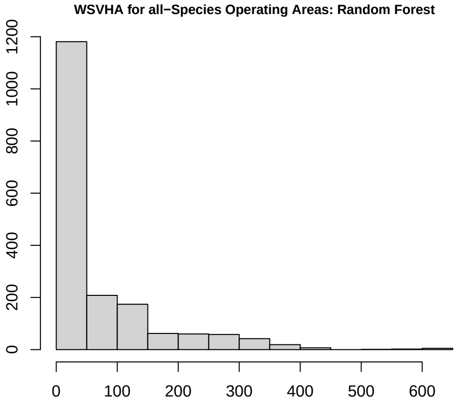
```
# visualize
rasterVis::levelplot(covs_m1_allSpeciesAreas)
```



2.7. All-Species Areas: Modeling WSVHA Estimates with Random Forest Regression

```
wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas <- raster::predict(covs_m1_allSpec
wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas$layer[wsvha_model1_randomForest_boo
writeRaster(wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas, overwrite=TRUE,
  filename = "/media/seamus/128GB_WORKD/data/raster/tcc/outputs/wsvha/bootstrapped/wsvha_model1_randomFo
plot(wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas, main="WSVHA for all-Species (
hist(wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas, main="WSVHA for all-Species (
```





3. Non-Fd Filtered Areas: Gaspard, Hawks Creek, and Meldrum Filtered by <50% Fd

3.1. Non-Fd Areas: Import DEM and CHM

```
# Merge chunks
filez_be_gaspard = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/BareEarth", full.nam
elev_raster_list_gaspard <- lapply(filez_be_gaspard, raster)
elev_raster_gaspard = do.call(merge, c(elev_raster_list_gaspard, tolerance = 1))
elev_rast_gaspard = terra::rast(elev_raster_gaspard)
terra::crs(elev_rast_gaspard) = "epsg:3005"
elev_rast_gaspard = terra::aggregate(elev_rast_gaspard, fact = 100, fun = mean)
writeRaster(elev_rast_gaspard, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/elev_ras

filez_be_hawks_creek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/BareEarth",
elev_raster_list_hawks_creek <- lapply(filez_be_hawks_creek, raster)
elev_raster_hawks_creek = do.call(merge, c(elev_raster_list_hawks_creek, tolerance = 1))
elev_rast_hawks_creek = terra::rast(elev_raster_hawks_creek)
terra::crs(elev_rast_hawks_creek) = "epsg:3005"
elev_rast_hawks_creek = terra::aggregate(elev_rast_hawks_creek, fact = 100, fun = mean)
writeRaster(elev_rast_hawks_creek, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/

filez_be_meldrum = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/BareEarth", full.nam
elev_raster_list_meldrum = lapply(filez_be_meldrum, raster)
elev_raster_meldrum = do.call(merge, c(elev_raster_list_meldrum, tolerance = 1))
elev_rast_meldrum = terra::rast(elev_raster_meldrum)
terra::crs(elev_rast_meldrum) = "epsg:3005"
elev_rast_meldrum = terra::aggregate(elev_rast_meldrum, fact = 100, fun = mean)
writeRaster(elev_rast_meldrum, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/elev_ras

filez_vh_gaspard = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/VegHt", full.names =
lead_htop_raster_list_gaspard <- lapply(filez_vh_gaspard, raster)
lead_htop_raster_gaspard = do.call(merge, c(lead_htop_raster_list_gaspard, tolerance = 1))
lead_htop_rast_gaspard = terra::rast(lead_htop_raster_gaspard)
terra::crs(lead_htop_rast_gaspard) = "epsg:3005"
lead_htop_rast_gaspard = terra::aggregate(lead_htop_rast_gaspard, fact = 100, fun = mean)
writeRaster(lead_htop_rast_gaspard, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/lea
filez_vh_hawks_creek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/VegHt", full
lead_htop_raster_list_hawks_creek <- lapply(filez_vh_hawks_creek, raster)
lead_htop_raster_hawks_creek = do.call(merge, c(lead_htop_raster_list_hawks_creek, tolerance = 1))
lead_htop_rast_hawks_creek = terra::rast(lead_htop_raster_hawks_creek)
terra::crs(lead_htop_rast_hawks_creek) = "epsg:3005"
lead_htop_rast_hawks_creek = terra::aggregate(lead_htop_rast_hawks_creek, fact = 100, fun = mean)
writeRaster(lead_htop_rast_hawks_creek, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_

filez_vh_meldrum = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/VegHt", full.names
lead_htop_raster_list_meldrum = list(filez_vh_meldrum, raster)
lead_htop_raster_meldrum = do.call(merge, c(lead_htop_raster_list_meldrum, tolerance = 1))
lead_htop_rast_meldrum = terra::rast(lead_htop_raster_meldrum)
terra::crs(lead_htop_rast_meldrum) = "epsg:3005"
lead_htop_rast_meldrum = terra::aggregate(lead_htop_rast_meldrum, fact = 100, fun = mean)
writeRaster(lead_htop_rast_meldrum, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/lea
# Merge mosaics
elev_raster_nonFdAreas_list = list(elev_raster_gaspard, elev_raster_hawks_creek, elev_raster_meldrum)
elev_raster_nonFdAreas = do.call(merge, c(elev_raster_nonFdAreas_list, tolerance = 1))
elev_rast_nonFdAreas = terra::rast(elev_raster_nonFdAreas)
writeRaster(elev_raster_nonFdAreas, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/elev_raste

lead_htop_raster_nonFdAreas_list = list(lead_htop_raster_gaspard, lead_htop_raster_hawks_creek, lead_ht
lead_htop_raster_nonFdAreas = do.call(merge, c(lead_htop_raster_nonFdAreas_list, tolerance = 1))
lead_htop_rast_nonFdAreas = terra::rast(lead_htop_raster_nonFdAreas)
writeRaster(lead_htop_raster_nonFdAreas, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/elevl
```