

Quesnel WSVHA (including validation cutblocks)

Cabin-GIS

24/06/2022

Contents

Action:	1
1. Quesnel Lake: Includes areas for all species (Fd and non-Fd stands), and developed cutblocks for validation purposes	2
1.1. Quesnel Lake: Import DEM & CHM	2
1.2. Quesnel Lake: Derive Terrain Rasters	2
1.3. Quesnel Lake: Derive Species Raster	3
1.4. Quesnel Lake: Derive Mask to Include Estimates of “RESULTS, TCC_Blocks” Areas . .	4
1.5. Quesnel Lake: Apply Masking	5
1.6. Quesnel Lake: Stack and Tidy Covariates	6
1.7. Quesnel Lake: Bootstrapping and training/test splits	7
1.8. Quesnel Lake: Modeling WSVHA Estimates with Random Forest Regression	9
2. All-Species Areas: Includes Fd and non-Fd stands	9
2.1. All-Species Areas: Import DEM and CHM	9
2.2. All-Species Areas: Derive Terrain Rasters	14
2.3. All-Species Areas: Derive Species Raster	15
2.4. All-Species Areas: Derive Mask From Approved Layers & Extents	16
2.5. All-Species Areas: Apply Masking	16
2.6. All-Species Areas: Stack and Tidy Covariates	17
2.7. All-Species Areas: Modeling WSVHA Estimates with Random Forest Regression	18
3. Non-Fd Filtered Areas: NEED CLARIFICATION	19
3.1. Non-Fd Areas:	19

Action:

The following report provides three outputs that were requested by LQ on June 24th. These included new rasters of WSVHA estimates for all 16 TCC Operating Areas that were updated with new masking protocols between areas of Fd-stands and nonFd-stands, and one additional test raster for Quesnel Lake that included validation cutblocks with cruise data available. To confirm details of the requested raster designs, these outputs and the report below were designed and numbered according to descriptions copied from email request and pasted below:

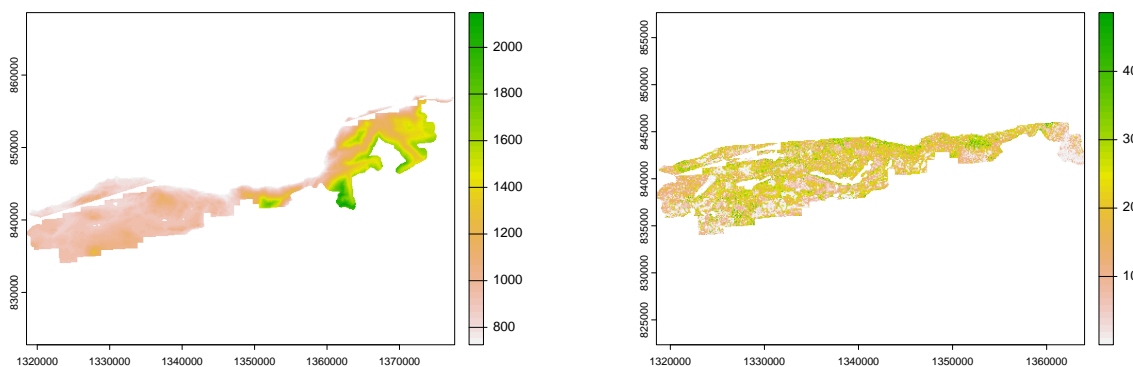
1. “All species, no removal (mask out) of developed blocks (RESULTS, TCC_Blocks): Quesnel Lake only”
2. “All species: All areas except Fd-areas (Gaspard, Hawks, Meldrum)”
3. “All species, standard mask (excluding developed blocks): All areas except Fd-areas (Gaspard, Hawks, Meldrum)”

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.names = TRUE)
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_rast_quesnel.tif", overwrite = TRUE)
elev_raster_quesnel = raster::raster("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/elev_raster_quesnel.tif")
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 = TRUE)
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/lead_htop_rast_quesnel.tif", overwrite = TRUE)
lead_htop_raster_quesnel = raster::raster("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/quesnel/lead_htop_raster_quesnel.tif")
lead_htop_rast_quesnel = terra::rast(lead_htop_raster_quesnel)
```

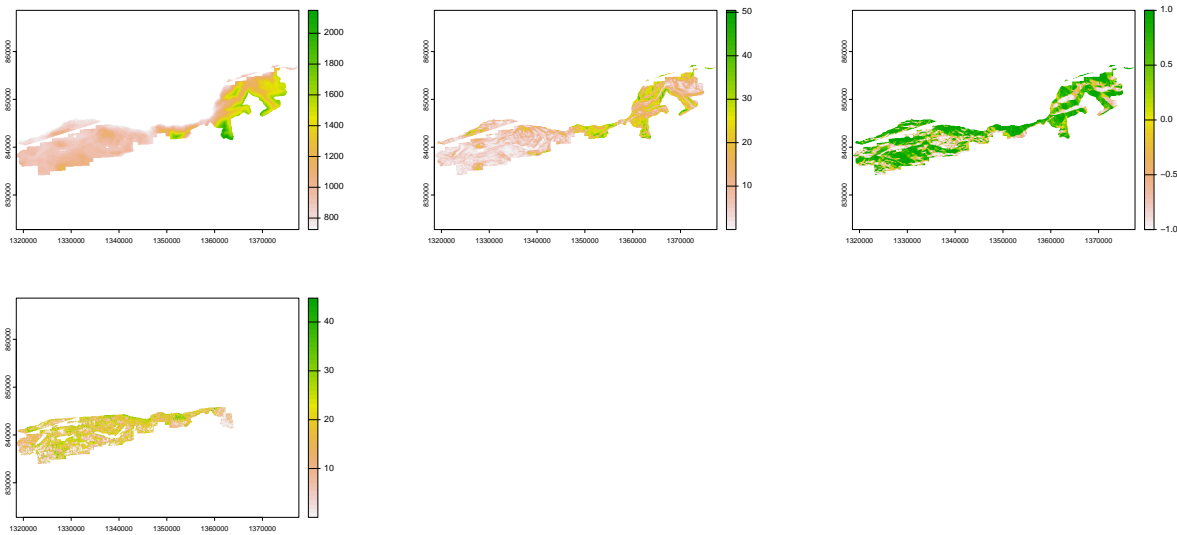


1.2. Quesnel Lake: Derive Terrain Rasters

```

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-co
writeRaster(asp_cos_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked-co
writeRaster(asp_sin_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked-co
writeRaster(lead_htop_rast_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unmasked-co

```



1.3. Quesnel Lake: Derive Species Raster

```

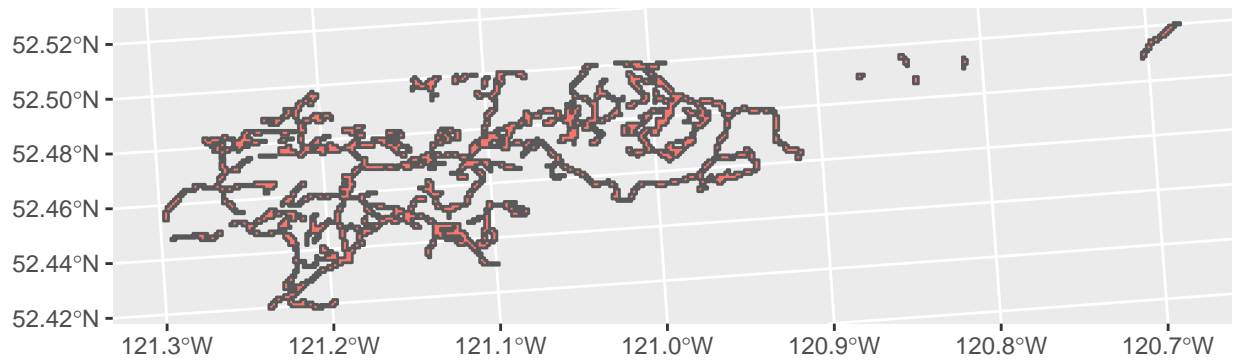
lead_htop_sv_quesnel = as.polygons(lead_htop_rast_quesnel)
lead_htop_sf_quesnel = sf::st_as_sf(lead_htop_sv_quesnel)
vri_sf = read_sf("/media/seamus/128GB_WORKD/data/vector/vri/vri_bc_2020_rank1.shp")
vri_species = vri_sf[c("SPECIES__1", "SPECIES_CD", "SPECIES_PC")]
vri_species_aoi = dplyr::filter(vri_species,
  SPECIES__1=='PL' | SPECIES__1=='PLI' | SPECIES__1=='FD' | SPECIES__1=='FDI' |
  SPECIES__1=='SB' | SPECIES__1=='SE' | SPECIES__1=='SW' | SPECIES__1=='SX' |
  SPECIES__1=='CW' | SPECIES__1=='HW' | SPECIES__1=='BL' | SPECIES__1=='LW')
vri_species_allspecies = vri_species_aoi
vri_species_allspecies$SPECIES__1 = dplyr::recode(vri_species_allspecies$SPECIES__1,
  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__1)
vri_species_allspecies = vri_species_allspecies["species_class"]
vri_species_allspecies_sf = sf::st_as_sf(vri_species_allspecies)
vri_species_aoi_ahbau = st_intersection(vri_species_allspecies_sf, st_make_valid(lead_htop_sf_quesnel))
species_class_rast_quesnel = terra::rasterize(vect(vri_species_aoi_ahbau), 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 TCC
mask_burn2018 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity TCC
mask_burn2021 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity TCC
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 = "ROUND")
mask_roads_ften = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/FTEN_Roads_All.shp")
mask_roads_ften = sf::st_zm(mask_roads_ften)
mask_roads_ften = sf::st_buffer(mask_roads_ften, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", joinStyle = "ROUND")
mask_clearcut = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/RSLT_CCRES_CLEAR.shp")
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_quesnel)
mask_burn2018_devBlocks_quesnel = sf::st_intersection(sf::st_make_valid(mask_burn2018), lead_htop_sf_quesnel)
mask_burn2021_devBlocks_quesnel = sf::st_intersection(sf::st_make_valid(mask_burn2021), lead_htop_sf_quesnel)
masks_df_devBlocks_quesnel = full_join(as_tibble(mask_burn2017_devBlocks_quesnel), as_tibble(mask_burn2018_devBlocks_quesnel), by = "id")
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_quesnel))
masks_df_devBlocks_quesnel = full_join(as_tibble(masks_sf_devBlocks_quesnel), as_tibble(mask_roads_devBlocks_quesnel), by = "id")
masks_sf_devBlocks_quesnel = st_as_sf(masks_df_devBlocks_quesnel)
masks_rast_devBlocks_quesnel = rasterize(vect(masks_sf_devBlocks_quesnel), lead_htop_rast_quesnel, touch = TRUE)
masks_raster_devBlocks_quesnel = raster::raster(masks_rast_devBlocks_quesnel)
writeRaster(masks_raster_devBlocks_quesnel, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/input_masks.tif", overwrite = TRUE)
ggplot(masks_sf_devBlocks_quesnel) + geom_sf(aes(fill = 'red'), show.legend = FALSE)
```



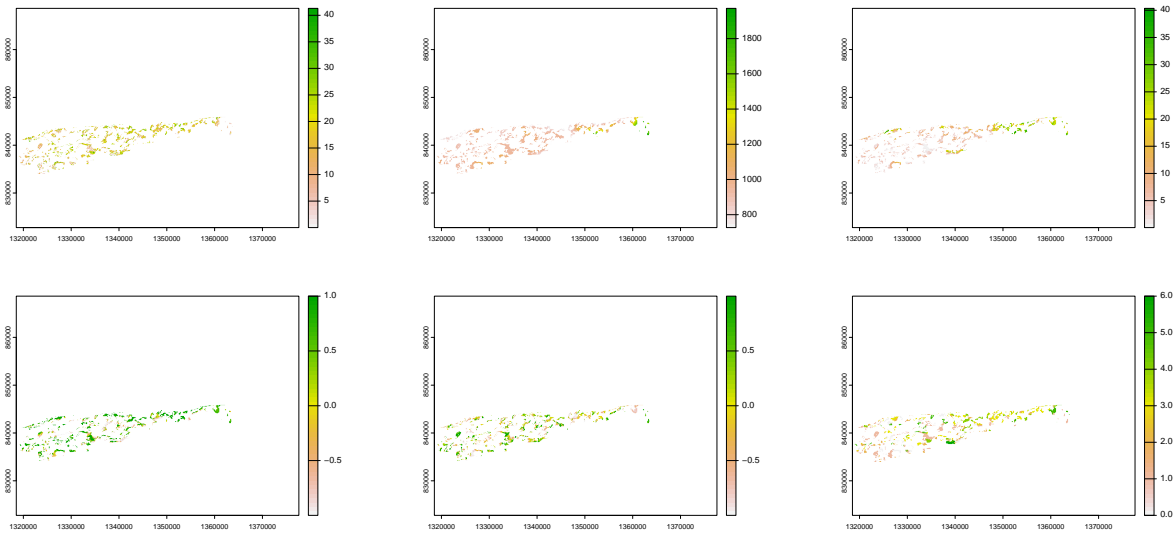
1.5. Quesnel Lake: Apply Masking

```
# 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, inverse=TRUE)

# mask by species
lead_htop_rast_quesnel_masked = mask(lead_htop_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FALSE)
elev_rast_quesnel_masked = mask(elev_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FALSE)
slope_rast_quesnel_masked = mask(slope_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FALSE)
asp_cos_rast_quesnel_masked = mask(asp_cos_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FALSE)
asp_sin_rast_quesnel_masked = mask(asp_sin_rast_quesnel_masked, species_class_rast_quesnel_masked, inverse=FALSE)

# save outputs
writeRaster(lead_htop_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/lead_htop_rast_quesnel_masked.tif")
writeRaster(elev_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/elev_rast_quesnel_masked.tif")
writeRaster(slope_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/slope_rast_quesnel_masked.tif")
writeRaster(asp_cos_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/asp_cos_rast_quesnel_masked.tif")
writeRaster(asp_sin_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/asp_sin_rast_quesnel_masked.tif")
```

```
writeRaster(species_class_rast_quesnel_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inq")
```



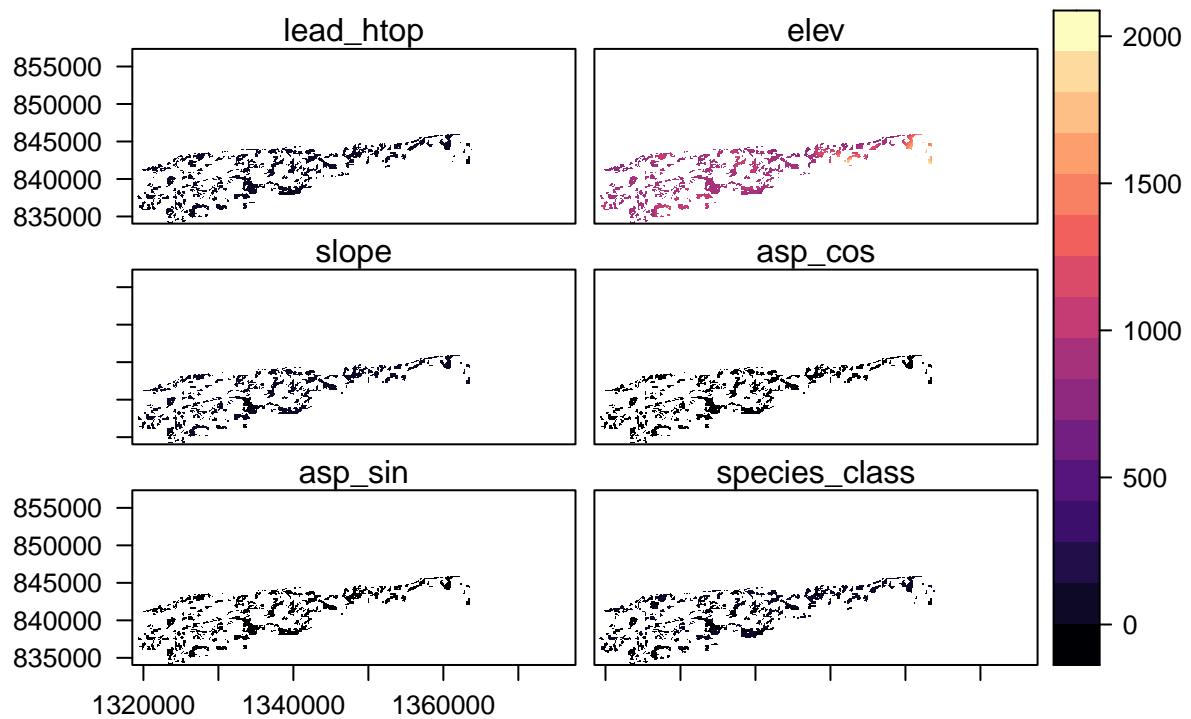
1.6. Quesnel Lake: Stack and Tidy Covariates

```
# 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
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

```
# Data cleaning
faib_psp <- read.csv("/media/seamus/128GB_WORKD/EFI-TCC/0_Caret_Predict_to_writeRasterOutput/Data/FAIB_LakeData/FAIB_LakeData.csv")
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)
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$swsha_L[faib_psp$swsha_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", "wsvha_L")]
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$lead_htop_rast_fullMask_allAreas_masked, adjust=0.8))
B = 1000
n = 4
faib_vri_true_m1_df_boot = dplyr::sample_n(faib_vri_true_m1_df, B * n, weight_by = dens.fun(faib_vri_true_m1_df$lead_htop))
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)")

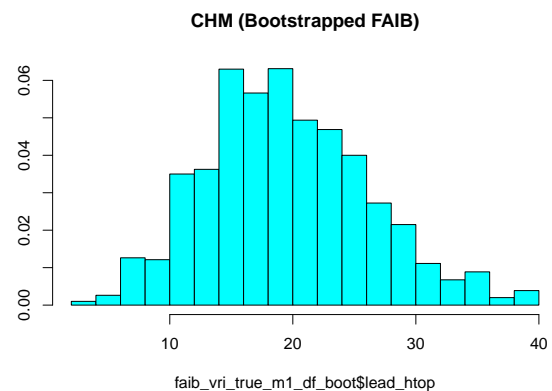
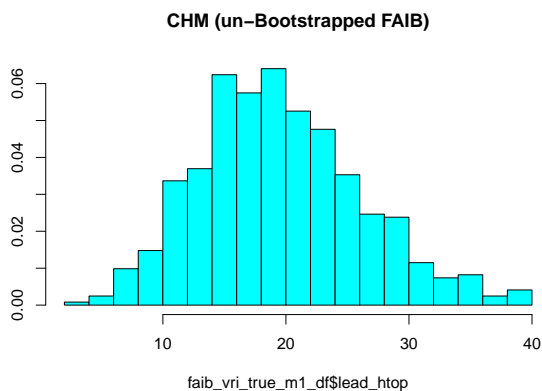
faib_vri_true_m1_df_boot_split = createDataPartition(faib_vri_true_m1_df_boot$wsvha_L, p=0.80, list=F)
faib_vri_true_m1_df_split = createDataPartition(faib_vri_true_m1_df$wsvha_L, p=0.80, list=F)

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]

```



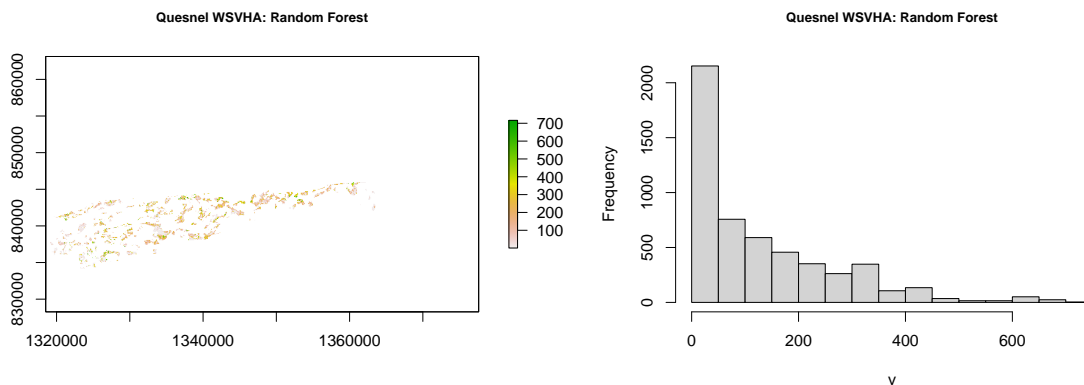
1.8. Quesnel Lake: Modeling WSVHA Estimates with Random Forest Regression

```
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_randomF")
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_quesnel,
wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks$layer[wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks$layer,
writeRaster(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, overwrite=TRUE,
  filename = "/media/seamus/128GB_WORKD/data/raster/tcc/outputs/wsvha/bootstrapped/wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks.tif")
plot(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, main="Quesnel WSVHA: Random Forest")
hist(wsvha_model1_randomForest_bootstrapped_demBased_100m_quesnelANDdevBlocks, main="Quesnel WSVHA: Random Forest")
```



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 = TRUE)
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_ahbau.tif")
```

```
filez_be_bells = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/bells/BareEarth", full.names = TRUE)
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_bells.tif")
```

```
filez_be_big_valley = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_valley/BareEarth", full.names = TRUE)
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/elev_raster_big_valley.tif")
```

```
filez_be_cariboo_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/cariboo_lake/BareEarth", full.names = TRUE)
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_lake/elev_raster_cariboo_lake.tif")
```

```
filez_be_charleson_marvincreek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/charleson_marvincreek/BareEarth", full.names = TRUE)
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 = 1))
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 = mean)
writeRaster(elev_rast_charleson_marvincreek, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/charleson_marvincreek/elev_raster_charleson_marvincreek.tif")
```

```
filez_be_dash = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/dash/BareEarth", full.names = TRUE)
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_dash.tif")
```

```
filez_be_little_river = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_river/BareEarth", full.names = TRUE)
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_river/elev_rast_little_river.tif")

filez_be_little_swift = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_swift/BareEarth", full.names = T)
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_swift/elev_rast_little_swift.tif")

filez_be_mcintosh = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/mcintosh/BareEarth", full.names = T)
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_rast_mcintosh.tif")

filez_be_phillips_anahim_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phillips_anahim_lake/BareEarth", full.names = T)
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/phillips_anahim_lake/elev_rast_phillips_anahim_lake.tif")

filez_be_piltz = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/piltz/BareEarth", full.names = T)
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_rast_piltz.tif")

filez_be_punky_clisbako = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punky_clisbako/BareEarth", full.names = T)
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_clisbako/elev_rast_punky_clisbako.tif")

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_htop_rast_ahbau.tif")

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_h

filez_vh_big_valley = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/big_valley/VegHt", full.
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_val

filez_vh_cariboo_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/cariboo_lake/VegHt", f
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/carib

filez_vh_charleson_marvincreek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/charleson_mar
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,
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/mosa

filez_vh_dash = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/dash/VegHt", full.names = T, a
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/l

filez_vh_little_river = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/little_river/VegHt", f
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", f
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 = T)
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/lead_h", overwrite = T)

filez_vh_phillips_anahim_lake = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phillips_anahim_lake/VegHt", full.names = T)
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, tolerance = 1))
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, fun = mean)
writeRaster(lead_htop_rast_phillips_anahim_lake, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/phillips_anahim_lake/lead_h", overwrite = T)

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"
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_h", overwrite = T)

filez_vh_punky_clisbako = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/punky_clisbako/VegHt", full.names = T)
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/punky_clisbako/lead_h", overwrite = T)

# 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_rast_allSpeciesAreas", overwrite = T)

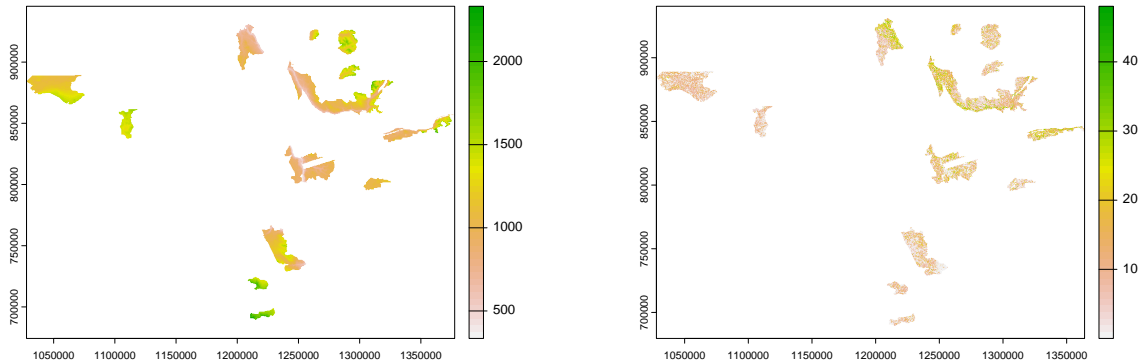
lead_htop_raster_allSpeciesAreas_list = list(
  lead_htop_raster_ahbau, #lead_htop_raster_bells,
  lead_htop_raster_big_valley, lead_htop_raster_cariboo_lake,
  lead_htop_raster_charleson_marvincreek, lead_htop_raster_dash,
  lead_htop_raster_little_river, lead_htop_raster_little_swift,
  lead_htop_raster_mcintosh, 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 = 0.001))
lead_htop_rast_allSpeciesAreas = terra::rast(lead_htop_raster_allSpeciesAreas)
writeRaster(lead_htop_raster_allSpeciesAreas, filename = "/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/lead_htop_rast_allSpeciesAreas.tif")

```

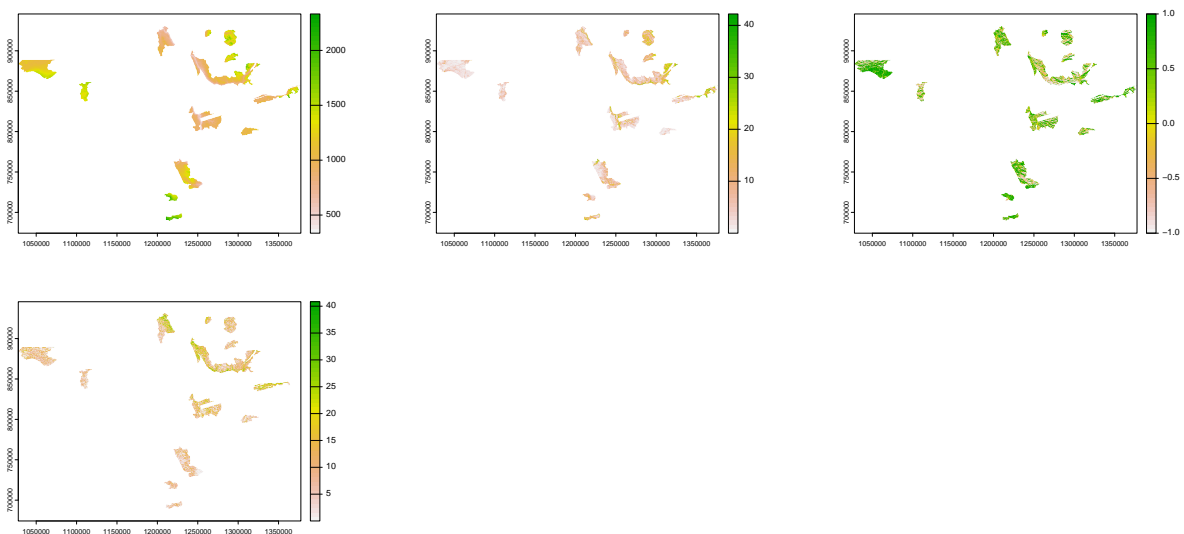


2.2. All-Species Areas: Derive Terrain Rasters

```

slope_rast_allSpeciesAreas = terra::terrain(elev_rast_allSpeciesAreas, v="slope", unit="degrees", neighbor="max")
aspect_rast_allSpeciesAreas = terra::terrain(elev_rast_allSpeciesAreas, v="aspect", unit="degrees", neighbor="max")
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_allSpeciesAreas)
writeRaster(elev_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm_rast_elev.tif")
writeRaster(slope_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm_rast_slope.tif")
writeRaster(asp_cos_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm_rast_osp.tif")
writeRaster(asp_sin_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm_rast_osi.tif")
writeRaster(lead_htop_rast_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/inputs/unm_rast_lead.tif")

```

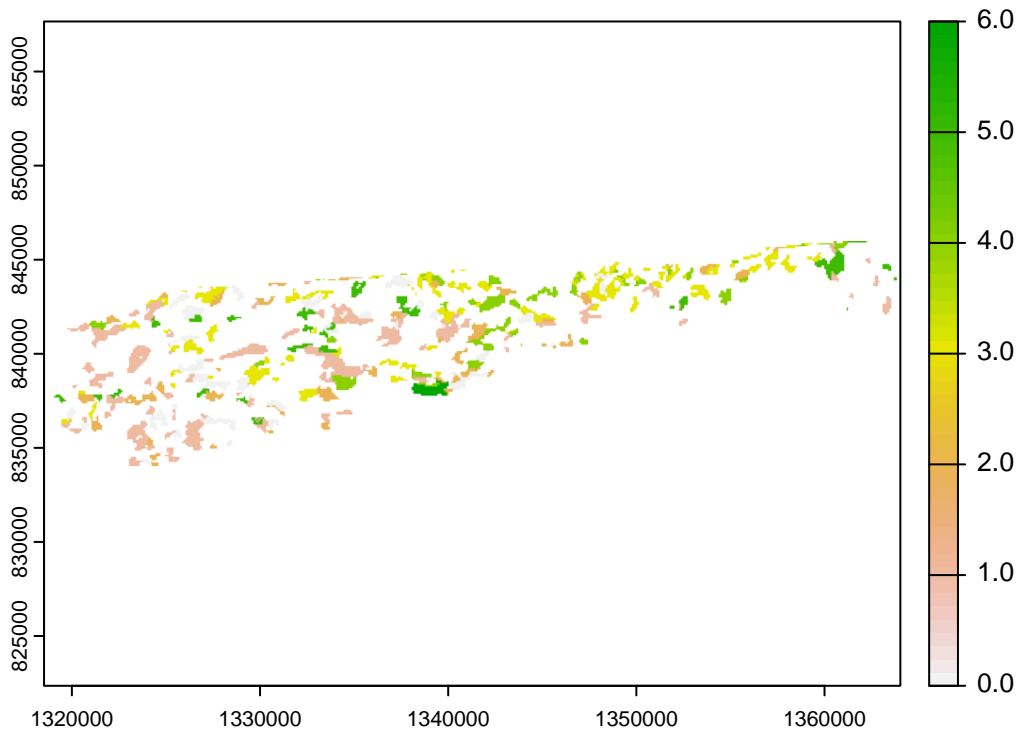


2.3. All-Species Areas: Derive Species Raster

```

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_species_allSpeciesAreas = vri_sf_allSpeciesAreas[c("SPECIES__1", "SPECIES_CD", "SPECIES_PC")]
vri_species_aoi_allSpeciesAreas = dplyr::filter(vri_species_allSpeciesAreas,
  SPECIES__1=='PL' | SPECIES__1=='PLI' | SPECIES__1=='FD' | SPECIES__1=='FDI' |
  SPECIES__1=='SB' | SPECIES__1=='SE' | SPECIES__1=='SW' | SPECIES__1=='SX' |
  SPECIES__1=='CW' | SPECIES__1=='HW' | SPECIES__1=='BL' | SPECIES__1=='LW')
vri_species_allSpeciesAreas = vri_species_aoi_allSpeciesAreas
vri_species_allSpeciesAreas$SPECIES__1 = dplyr::recode(vri_species_allSpeciesAreas$SPECIES__1,
  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__1)
vri_species_allSpeciesAreas = vri_species_allSpeciesAreas["species_class"]
vri_species_allSpeciesAreas_sf = sf::st_as_sf(vri_species_allSpeciesAreas)
vri_species_aoi_allSpeciesAreas = st_intersection(vri_species_allSpeciesAreas_sf, st_make_valid(lead_htop_rast_allSpeciesAreas))
species_class_rast_allSpeciesAreas = terra::rasterize(vect(vri_species_aoi_allSpeciesAreas), lead_htop_rast_allSpeciesAreas)
species_class_rast_allSpeciesAreas = terra::resample(species_class_rast_allSpeciesAreas, lead_htop_rast_allSpeciesAreas)
species_class_raster_allSpeciesAreas = raster::raster(species_class_rast_allSpeciesAreas)
raster::writeRaster(species_class_raster_allSpeciesAreas, filename = "/media/seamus/128GB_WORKD/data/species_class_raster_allSpeciesAreas.tif")

```



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 TCC
mask_burn2018 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity TCC
mask_burn2021 = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/TCC_Burn_Severity TCC
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 = "ROUND")
mask_roads_ften = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/FTEN_Roads_All.shp")
mask_roads_ften = sf::st_zm(mask_roads_ften)
mask_roads_ften = sf::st_buffer(mask_roads_ften, dist = 15, nQuadSegs = 5, endCapStyle = "ROUND", joinStyle = "ROUND")
mask_clearcut = sf::read_sf("/media/seamus/128GB_WORKD/data/vector/tcc_mask_layers/RSLT_CCRES_CLEAR.shp")
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_3005)
mask_burn2018_fullMask_allAreas = sf::st_intersection(sf::st_make_valid(mask_burn2018), extent_sfz_3005)
mask_burn2021_fullMask_allAreas = sf::st_intersection(sf::st_make_valid(mask_burn2021), extent_sfz_3005)
masks_df_fullMask_allAreas = full_join(as_tibble(mask_burn2017_fullMask_allAreas), as_tibble(mask_burn2018_fullMask_allAreas), by = "name")
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_fullMask_allAreas), by = "name")
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_fullMask_allAreas), by = "name")
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_fullMask_allAreas), by = "name")
masks_sf_fullMask_allAreas = st_as_sf(masks_df_fullMask_allAreas)
masks_rast_fullMask_allAreas = rasterize(vect(masks_sf_fullMask_allAreas), lead_htop_rast_all, touches = TRUE)
masks_raster_fullMask_allAreas = raster::raster(masks_rast_fullMask_allAreas)
writeRaster(masks_raster_fullMask_allAreas, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/input_masks_raster.shp")
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_allSpeciesAreas)
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_allSpeciesAreas)
```



```

# mask by mask
lead_htop_rast_allSpeciesAreas_masked = mask(lead_htop_rast_allSpeciesAreas, masks_rast_fullMask_allAreas)
elev_rast_allSpeciesAreas_masked = mask(elev_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inverse=TRUE)
slope_rast_allSpeciesAreas_masked = mask(slope_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inverse=TRUE)
asp_cos_rast_allSpeciesAreas_masked = mask(asp_cos_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inverse=TRUE)
asp_sin_rast_allSpeciesAreas_masked = mask(asp_sin_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inverse=TRUE)
species_class_rast_allSpeciesAreas_masked = mask(species_class_rast_allSpeciesAreas, masks_rast_fullMask_allAreas, inverse=TRUE)

# mask by species
lead_htop_rast_allSpeciesAreas_masked = mask(lead_htop_rast_allSpeciesAreas_masked, species_class_rast_allSpeciesAreas_masked)
elev_rast_allSpeciesAreas_masked = mask(elev_rast_allSpeciesAreas_masked, species_class_rast_allSpeciesAreas_masked)
slope_rast_allSpeciesAreas_masked = mask(slope_rast_allSpeciesAreas_masked, species_class_rast_allSpeciesAreas_masked)
asp_cos_rast_allSpeciesAreas_masked = mask(asp_cos_rast_allSpeciesAreas_masked, species_class_rast_allSpeciesAreas_masked)
asp_sin_rast_allSpeciesAreas_masked = mask(asp_sin_rast_allSpeciesAreas_masked, species_class_rast_allSpeciesAreas_masked)
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/lead_htop_rast_allSpeciesAreas_masked.tif")
writeRaster(elev_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/elev_rast_allSpeciesAreas_masked.tif")
writeRaster(slope_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/slope_rast_allSpeciesAreas_masked.tif")
writeRaster(asp_cos_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/asp_cos_rast_allSpeciesAreas_masked.tif")
writeRaster(asp_sin_rast_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/asp_sin_rast_allSpeciesAreas_masked.tif")
writeRaster(species_class_raster_allSpeciesAreas_masked, filename = "/media/seamus/128GB_WORKD/data/raster/tcc/species_class_raster_allSpeciesAreas_masked.tif")

```

2.6. All-Species Areas: Stack and Tidy Covariates

```

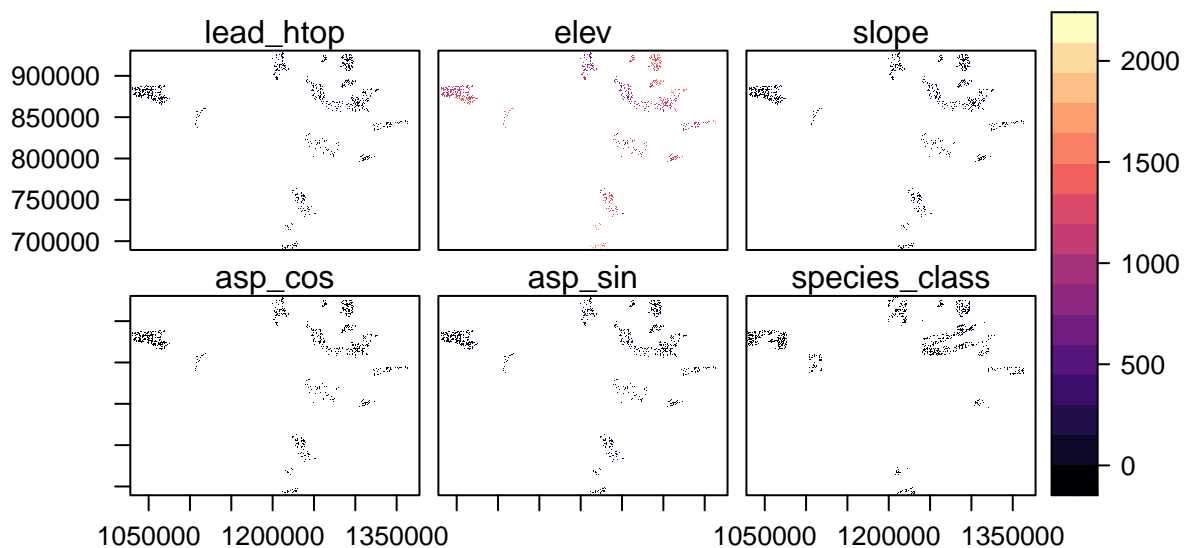
# Rename rasters
names(lead_htop_rast_allSpeciesAreas_masked) = "lead_htop"
names(elev_rast_allSpeciesAreas_masked) = "elev"
names(slope_rast_allSpeciesAreas_masked) = "slope"
names(asp_cos_rast_allSpeciesAreas_masked) = "asp_cos"
names(asp_sin_rast_allSpeciesAreas_masked) = "asp_sin"
names(species_class_rast_allSpeciesAreas_masked) = "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)

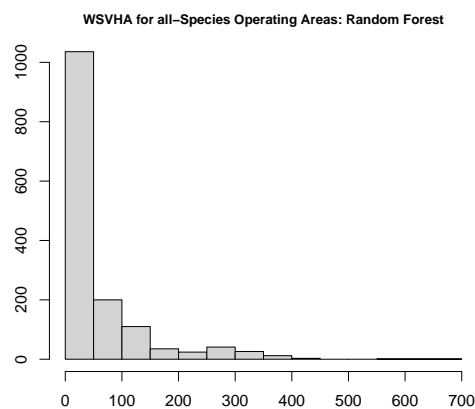
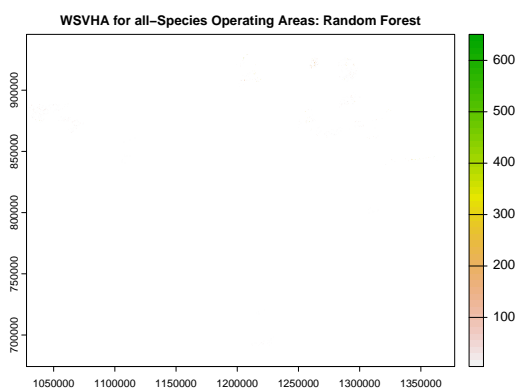
# visualize
rasterVis::levelplot(covs_m1_allSpeciesAreas)

```



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_bo
writeRaster(wsvha_model1_randomForest_bootstrapped_demBased_100m_allSpeciesAreas, overwrite=TRUE,
  filename = "/media/seamus/128GB_WORKKD/data/raster/tcc/outputs/wsvha/bootstrapped/wsvha_model1_randomF
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: NEED CLARIFICATION

3.1. Non-Fd Areas:

```
# Merge chunks
filez_be_gaspard = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/BareEarth", full.names = TRUE)
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_rast_gaspard.tif")

filez_be_hawks_creek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/BareEarth", full.names = TRUE)
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/elev_rast_hawks_creek.tif")

filez_be_meldrum = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/BareEarth", full.names = TRUE)
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_rast_meldrum.tif")

filez_vh_gaspard = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/gaspard/VegHt", full.names = TRUE)
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/lead_htop_rast_gaspard.tif")

filez_vh_hawks_creek = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/hawks_creek/VegHt", full.names = TRUE)
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_creek/lead_htop_rast_hawks_creek.tif")

filez_vh_meldrum = list.files("/media/seamus/Ubuntu 22_04 LTS amd64/mosaics/meldrum/VegHt", full.names = TRUE)
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/lead_htop_rast_meldrum.tif")

# 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_raster")

lead_htop_raster_nonFdAreas_list = list(lead_htop_raster_gaspard, lead_htop_raster_hawks_creek, lead_htop_raster_meldrum)
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/lead_htop_raster")

```