Spatialtemporal Analysis in R

Terrain Analysis

loading libraries

```
library(raster)
```

```
## Warning: package 'raster' was built under R version 3.4.3
```

```
## Loading required package: sp
```

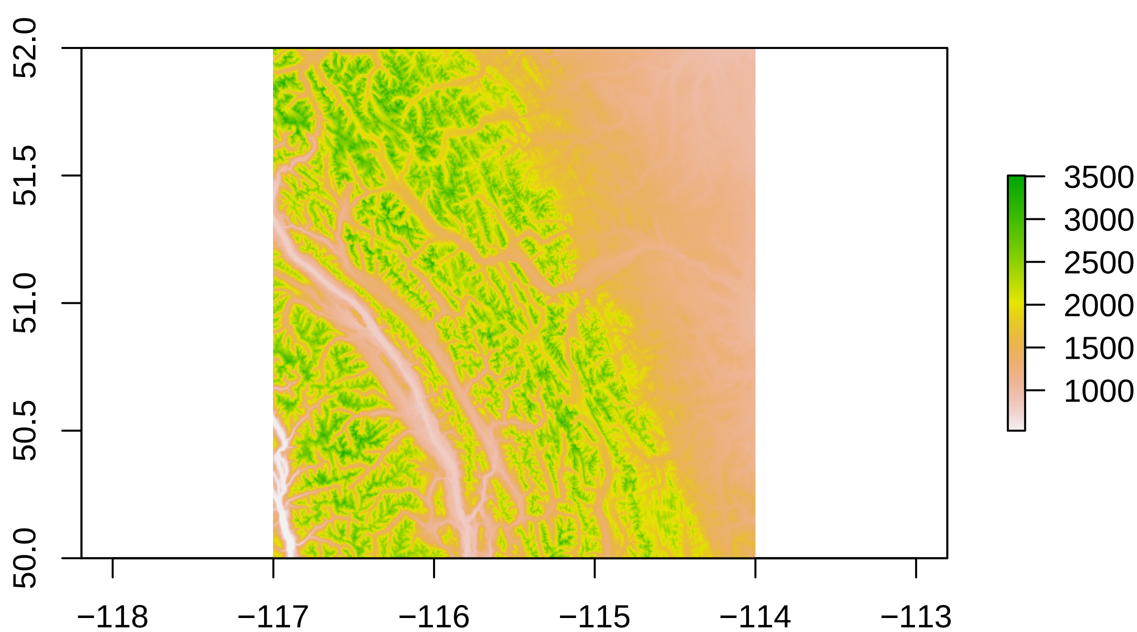
```
library(rgeos)
```

```
## Warning: package 'rgeos' was built under R version 3.4.3
```

```
## rgeos version: 0.3-26, (SVN revision 560)
##  GEOS runtime version: 3.6.1-CAPI-1.10.1 r0
##  Linking to sp version: 1.2-7
##  Polygon checking: TRUE
```

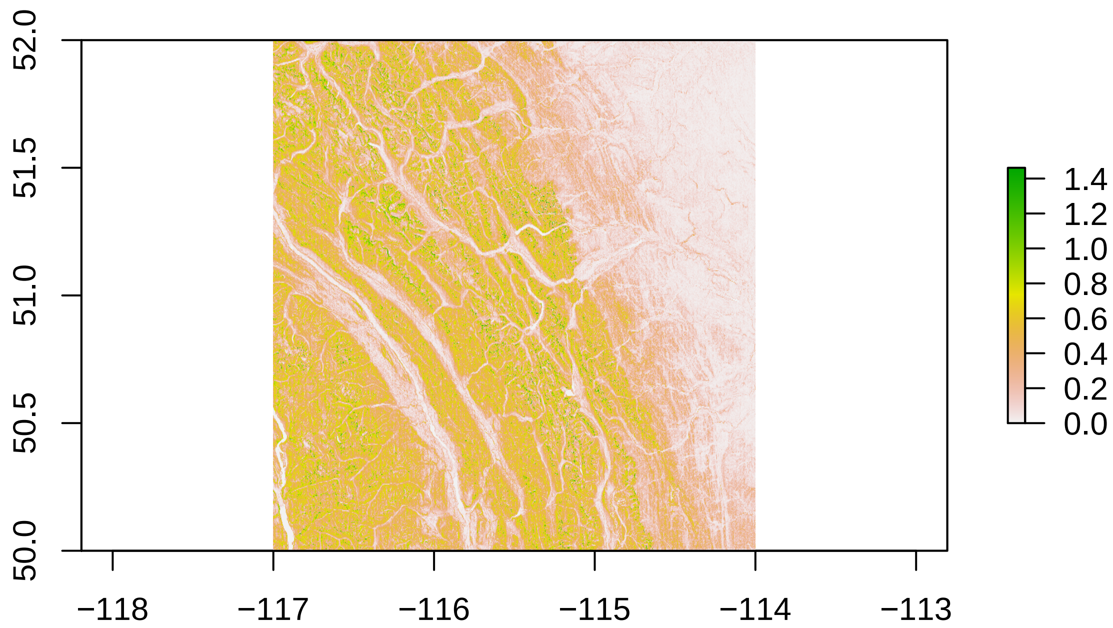
Ploting a raster

```
setwd("C:/Users/alb818/Dropbox/DATA/R_SPATIAL")

SRTM <- raster("SRTM_MOSAIC_NOVOID.tif")

plot(SRTM)
```



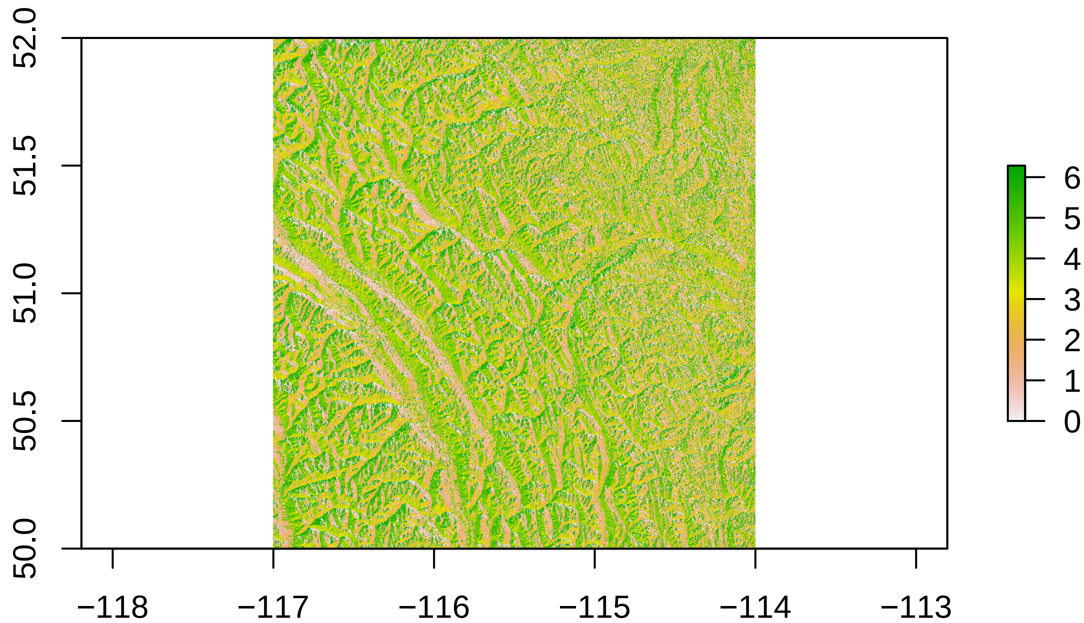
Calculate slope

```r
slope <- terrain(SRTM, opt = "slope", unit = "radians")

plot(slope)
```



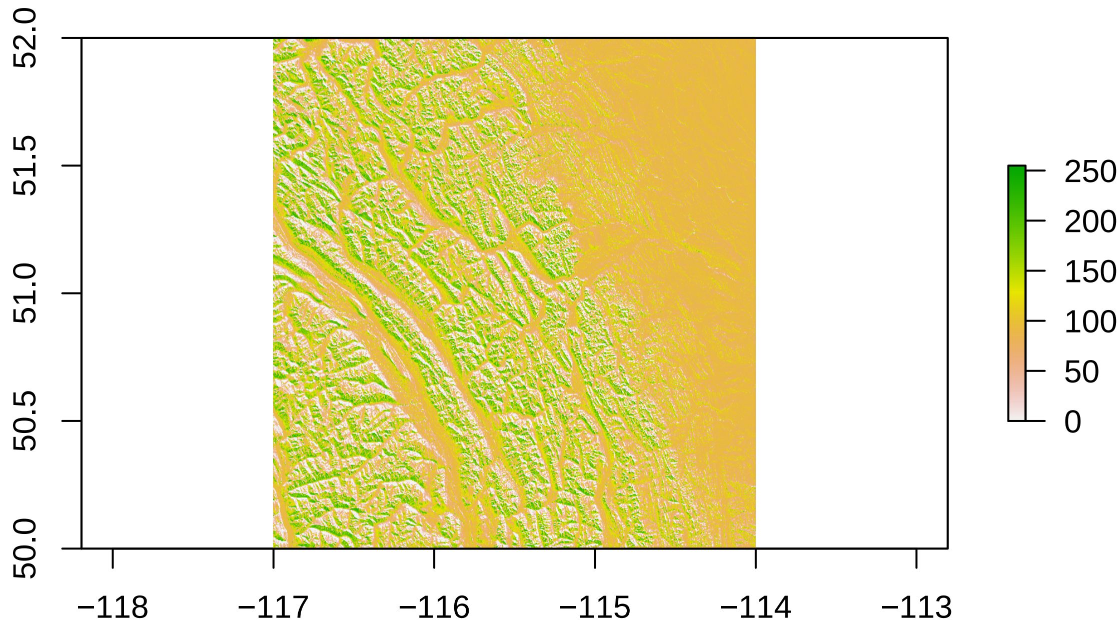
Calculate aspect

```r
aspect <- terrain(SRTM, opt = "aspect", unit = "radians")

plot(aspect)
```



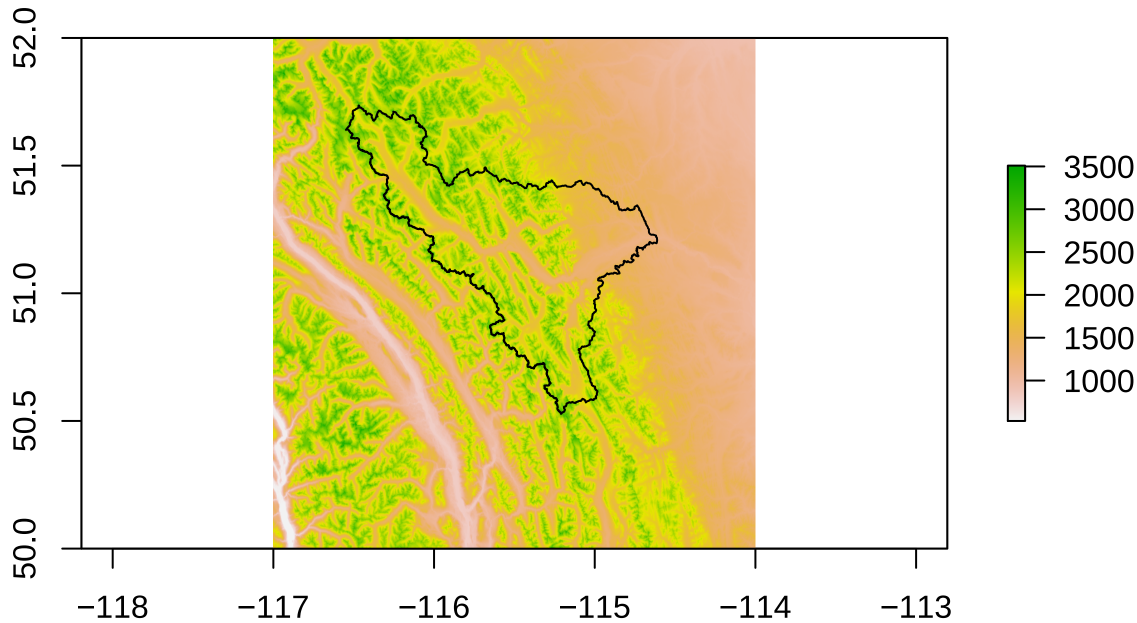
Calculate hillshade for today at noon

```
hillshd <- hillShade(slope, aspect, angle = 21.3, direction = 202.8, normalize = T)

plot(hillshd)
```



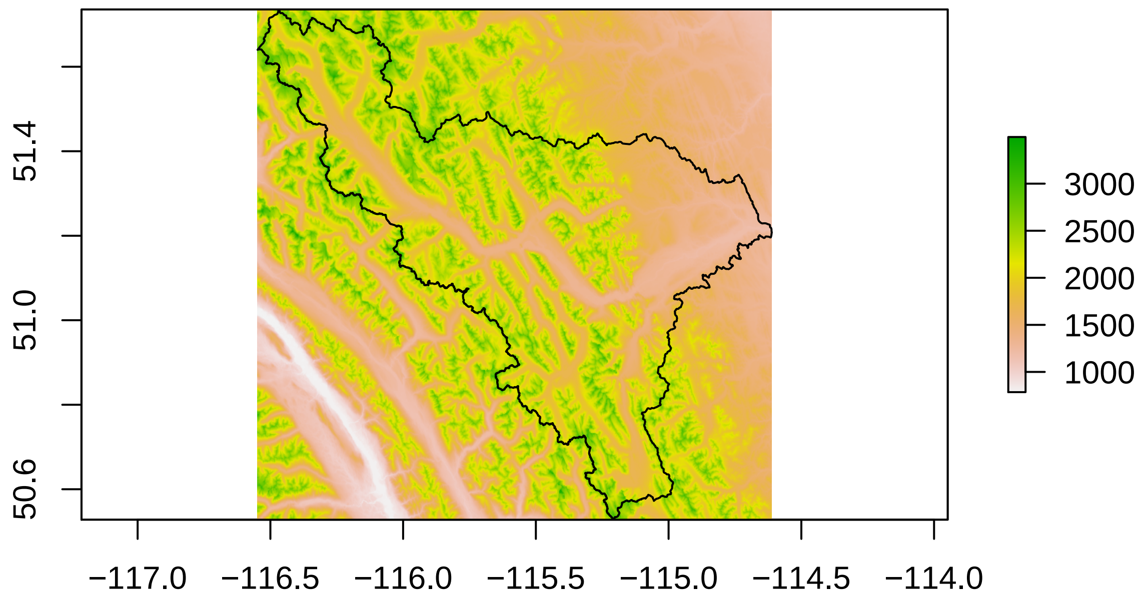
Loading and ploting a shapefile over a raster

```
BowRiver_atCochrane <- shapefile("BowRiver_atCochraneWGS84.shp")

plot(SRTM)
plot(BowRiver_atCochrane, add = T, bg = "transparent", border = "black")
```



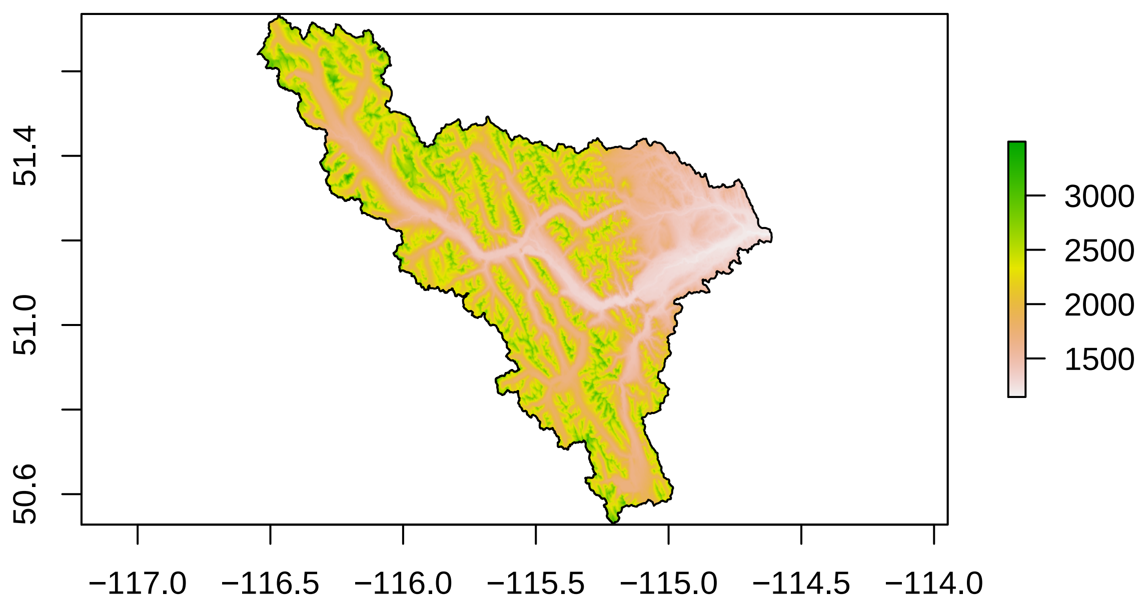
Croping a raster using the shapefile extent

```
AOI <- extent(BowRiver_atCochrane)

SRTM_BowBasin <- crop(SRTM, AOI)

plot(SRTM_BowBasin)
plot(BowRiver_atCochrane, add = T, bg = "transparent", border = "black")
```



masking a raster using a shapefile (same as clipping)

```
SRTM_BowBasin_mask <- mask(SRTM_BowBasin, BowRiver_atCochrane)

plot(SRTM_BowBasin_mask)
plot(BowRiver_atCochrane, add = T, bg = "transparent", border = "black")
```



Create HRUs based on aspect (reclassify)

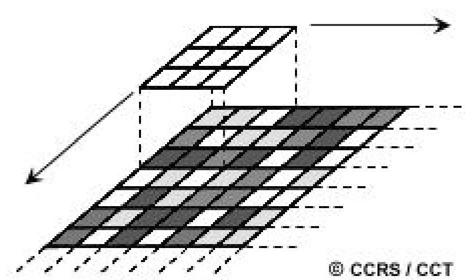
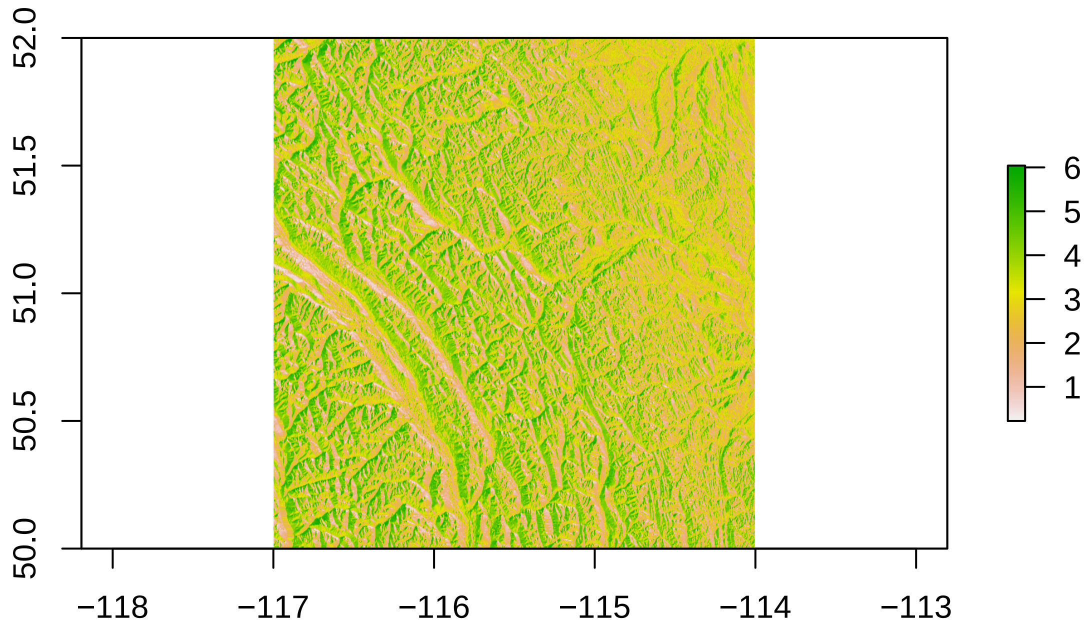
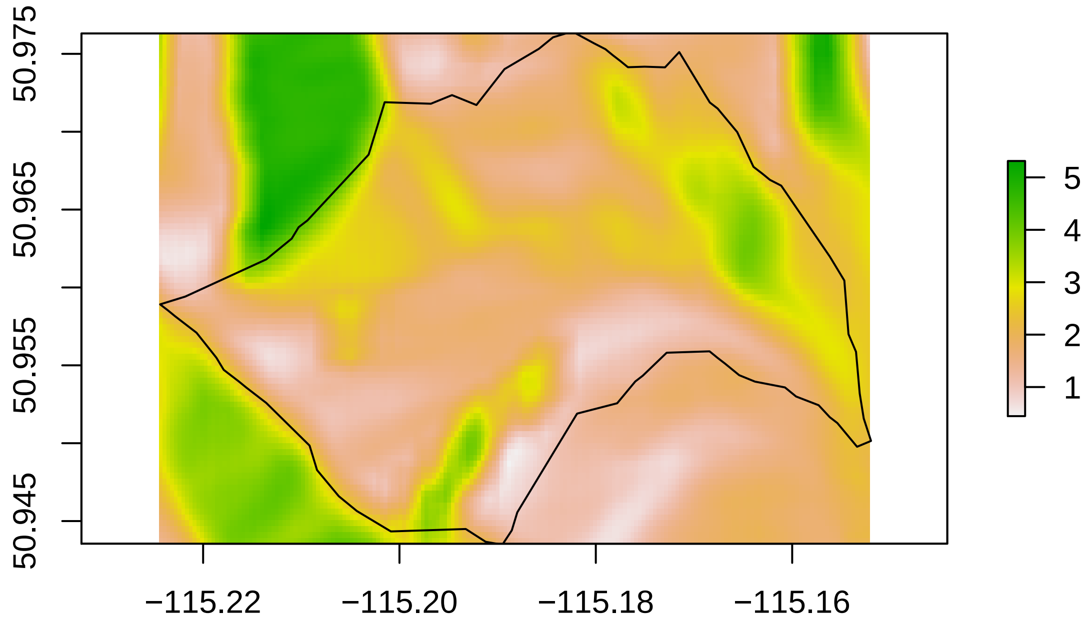


Figure 1: Spatial Filtering (Canada Centre for Remote Sensing, 2015)

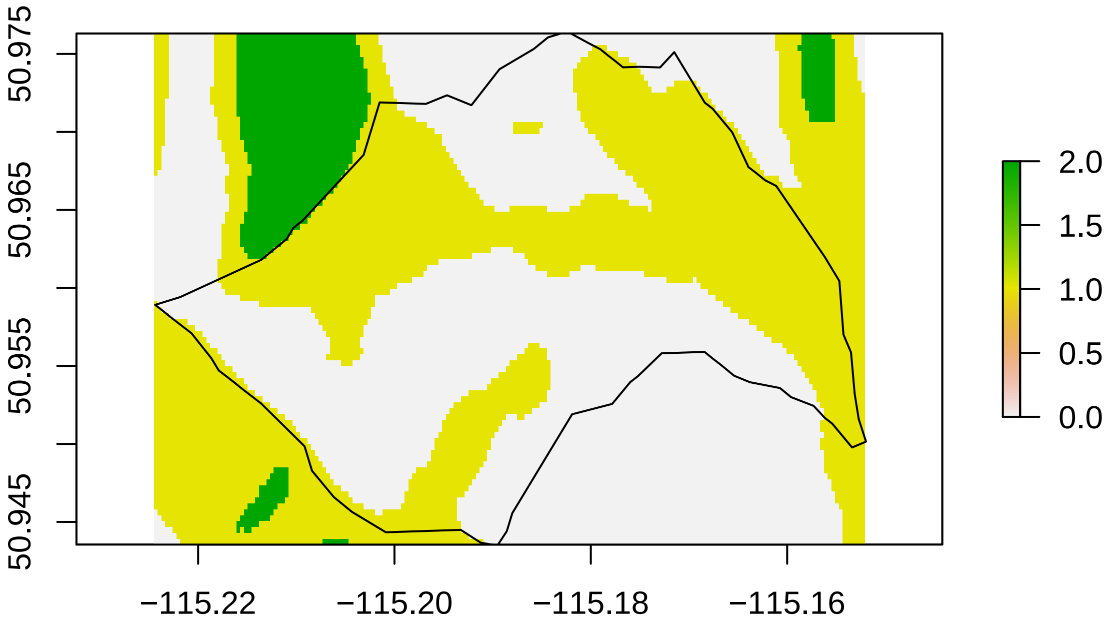
```
MarmotBasin <- shapefile("Marmot_Basin2.shp")

aspect_filter <- focal(aspect, w=matrix(1/121,nrow=11,ncol=11))

plot(aspect_filter)
```



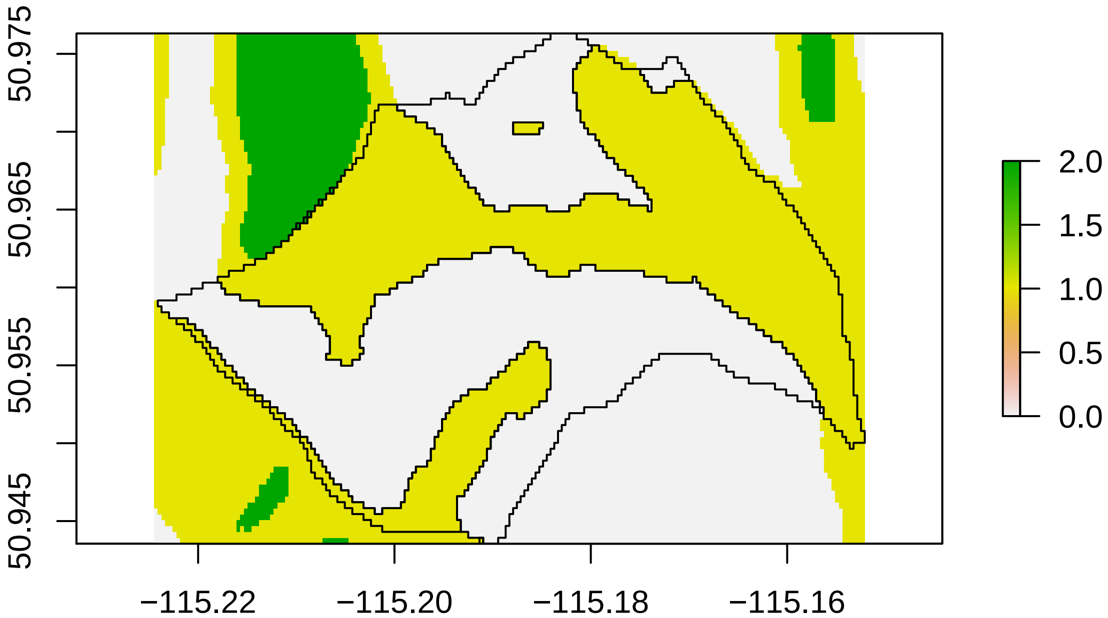

```
Marmot_Aspect <- crop(aspect_filter, MarmotBasin)

plot(Marmot_Aspect)
plot(MarmotBasin, add = T, bg = "transparent")
```




```r
B <- matrix(c(0, 2, 4, 6, 2, 4, 6, 7, 0, 1, 2, 3), nr = 4, nc = 3)

reclass_aspect <- reclassify(Marmot_Aspect, B)

plot(reclass_aspect)
plot(MarmotBasin, add = T, bg = "transparent")
```




```
reclass_aspect_mask <- mask(reclass_aspect, MarmotBasin)

HRUs <- rasterToPolygons(reclass_aspect_mask, dissolve = TRUE)
plot(reclass_aspect)
plot(HRUs, add = T, bg = "transparent")
```



Extracting slope values from HRUs

```
HRUs_slope <- as.matrix(extract(slope, HRUs))

HRU1 <- mean(HRUs_slope[[1]])*57.2958
HRU2 <- mean(HRUs_slope[[2]])*57.2958
HRU3 <- mean(HRUs_slope[[3]])*57.2958

print(HRU1)
```

```
## [1] 20.26551
```

```
print(HRU2)
```

```
## [1] 19.52238
```

```
print(HRU3)
```

```
## [1] 12.95594
```

Optical Remote Sensing

Loading Sentinel 2A bands

```
SA2_list <- list.files(path = getwd(), pattern = "S2A_20171106_B.*.tif$")

blue <- raster(SA2_list[5])
green <- raster(SA2_list[6])
```

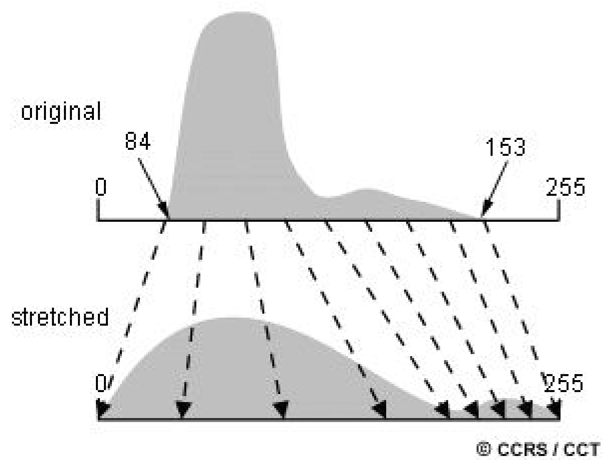
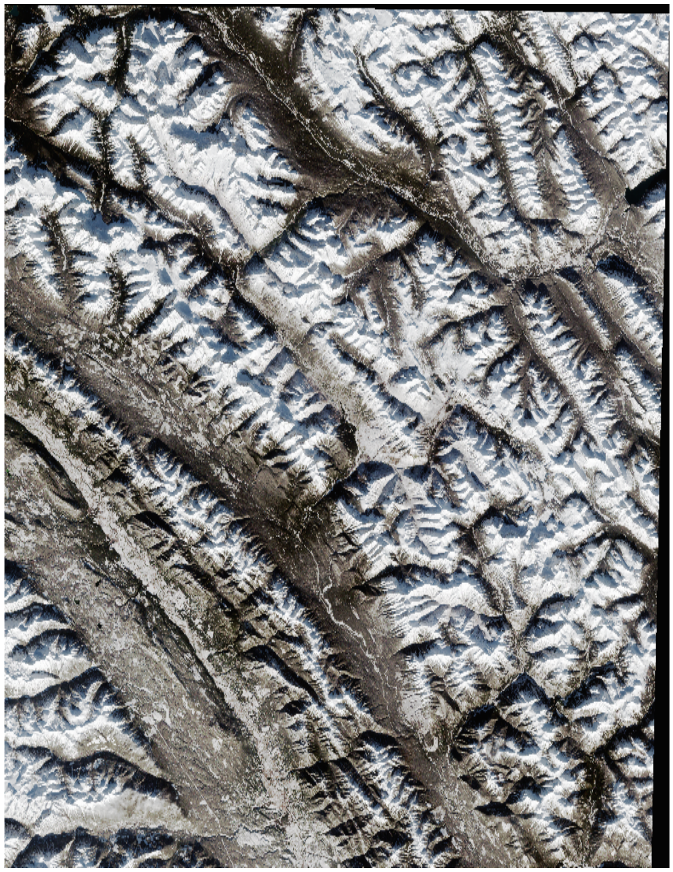


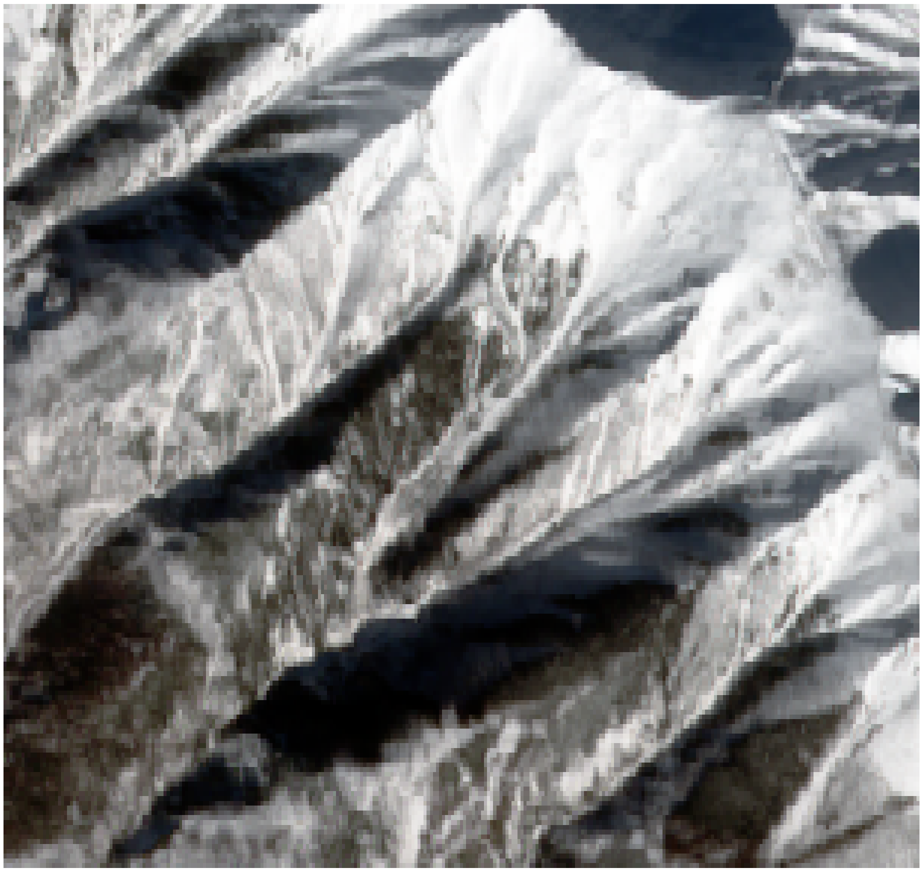
Figure 2: Histrogram Streching (CCRS, 2015)

```
red <- raster(SA2_list[7])
NIR <- raster(SA2_list[11])
SWIR_1610nm <- raster(SA2_list[3])

blue <- crop(blue, AOI)/77.53
green <- crop(green, AOI)/78.31
red <- crop(red, AOI)/90.45

RGB_data <- stack(blue, green, red)

plotRGB(RGB_data, 3, 2, 1, stretch = "hist")
```



Zoom in some extent

```
ext <- extent(-115.8905, -115.8559, 50.9709, 50.9914)

plotRGB(RGB_data, 3, 2, 1, stretch = "hist", ext = ext)
```



Calculate NDSI

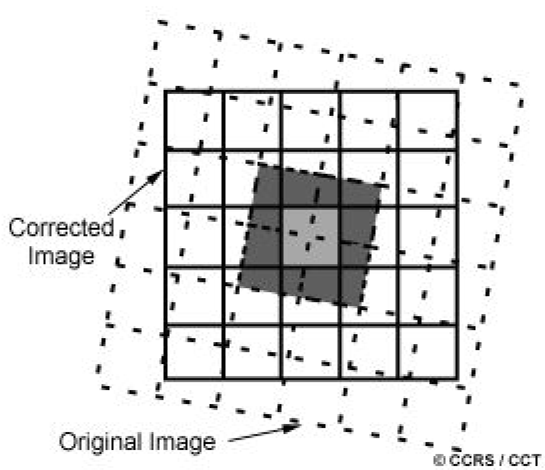
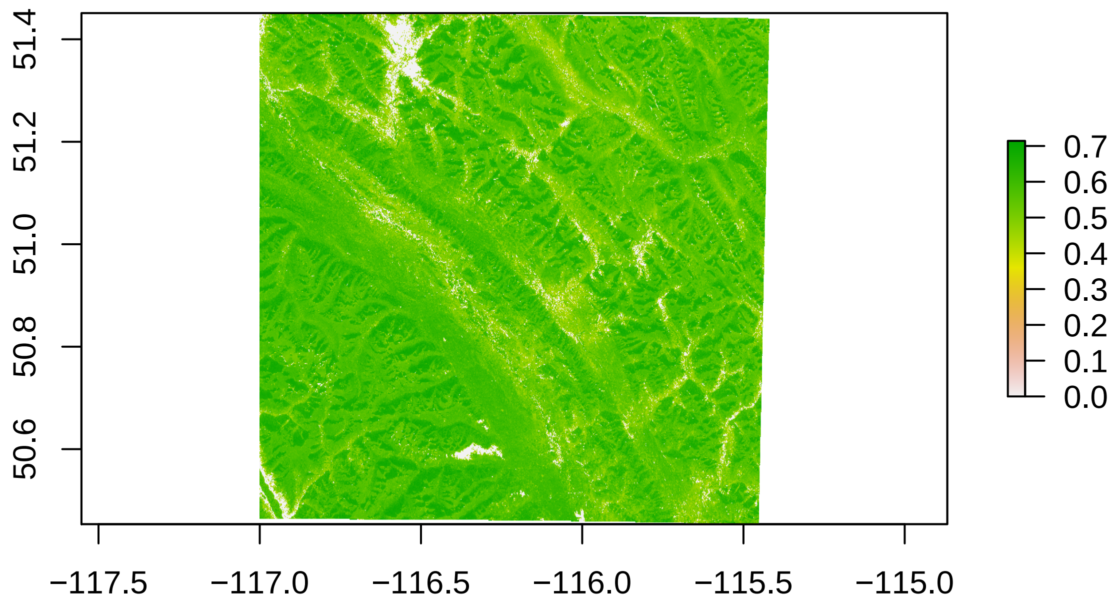


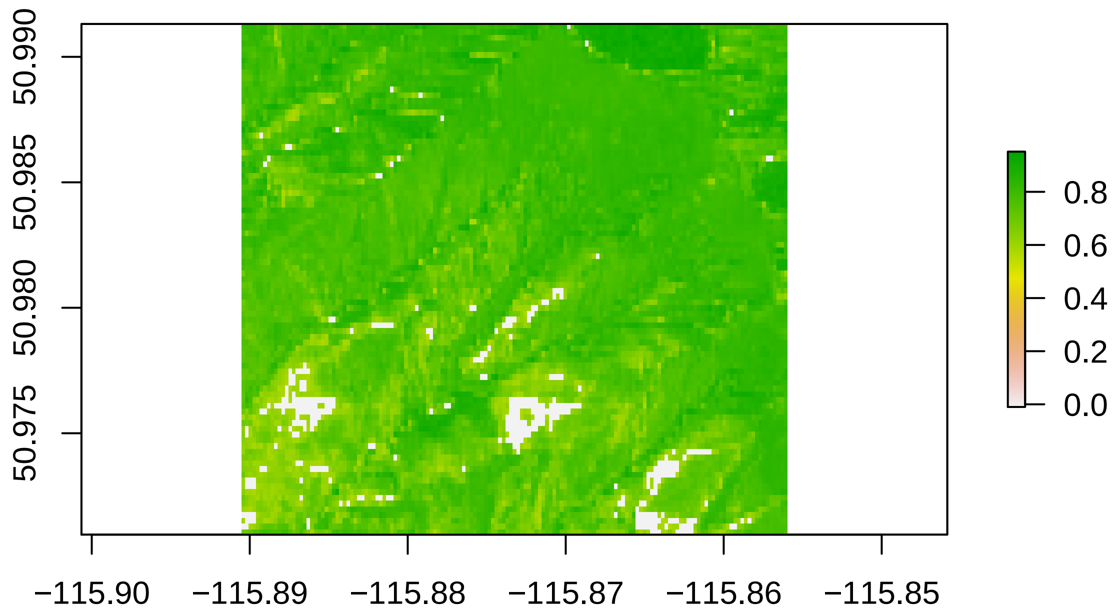
Figure 3: Spatial Resampling (CCRS, 2015)

```
NIR_resample <- resample(NIR, SWIR_1610nm)

NDSI <- (NIR_resample - SWIR_1610nm)/(NIR_resample + SWIR_1610nm)

NDSI <- NDSI/1.4
```

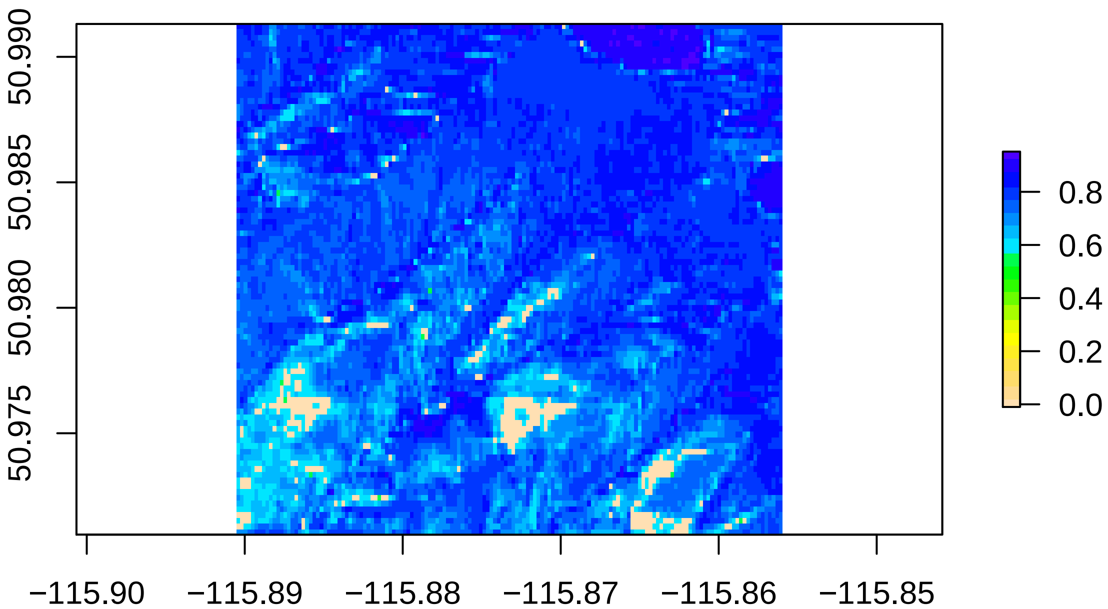
```r
NDSI[NDSI < 0.4] <- 0

plot(NDSI)
```




```r
fSCA <- -0.01 + 1.45*NDSI

plot(fSCA, ext = ext)
```

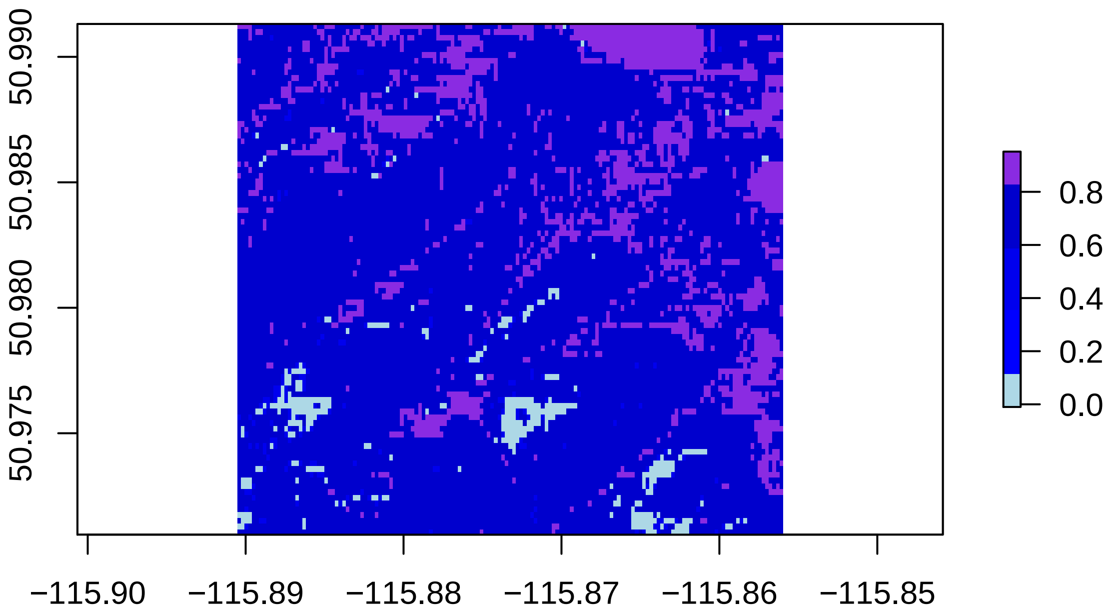



```
plot(fSCA, ext = ext, col = rev(topo.colors(20)))
```



Make your own colour pallet

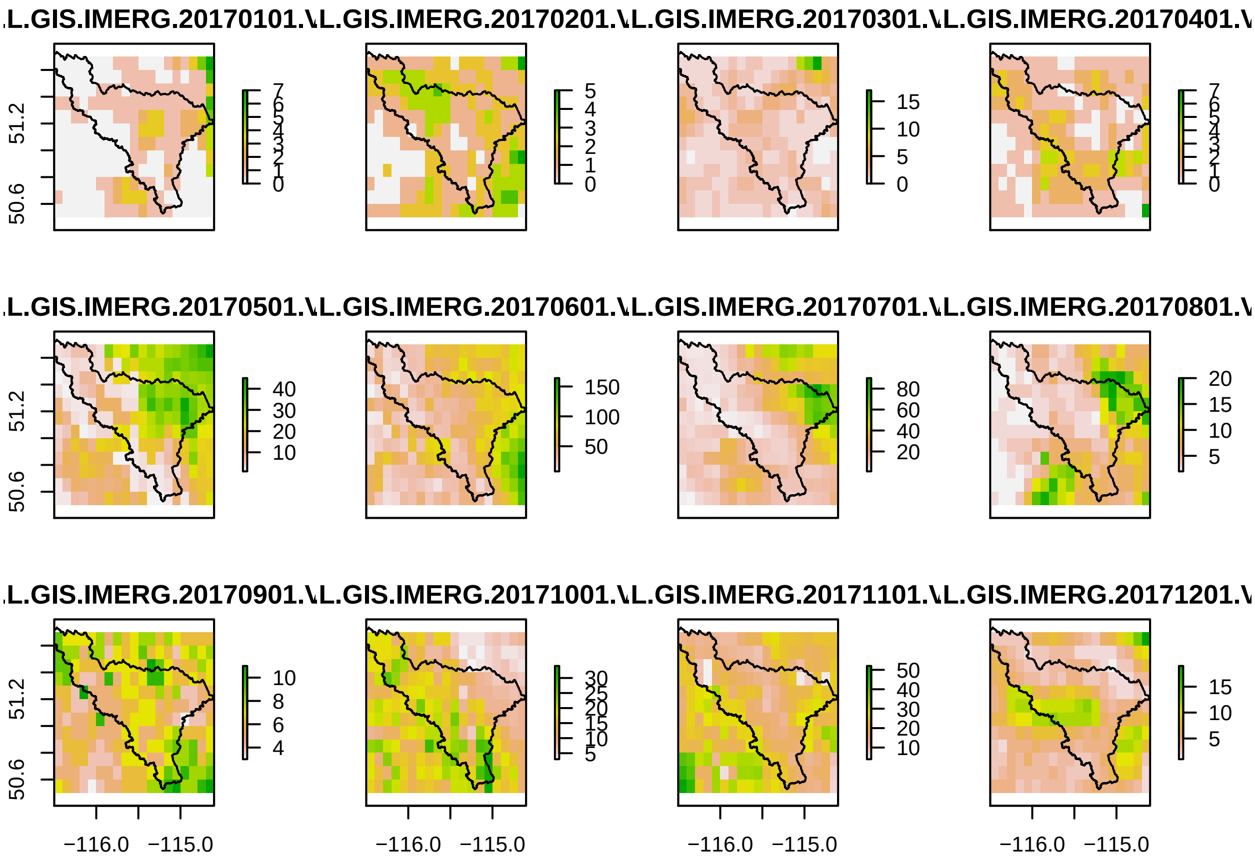
```
breakpoints <- c(0.2, 0.4, 0.6, 0.8, 1)
colors <- c("lightblue", "blue", "blue2", "blue3", "blueviolet")

plot(fSCA, ext = ext, col = colors)
```



Spatiotemporal Analysis

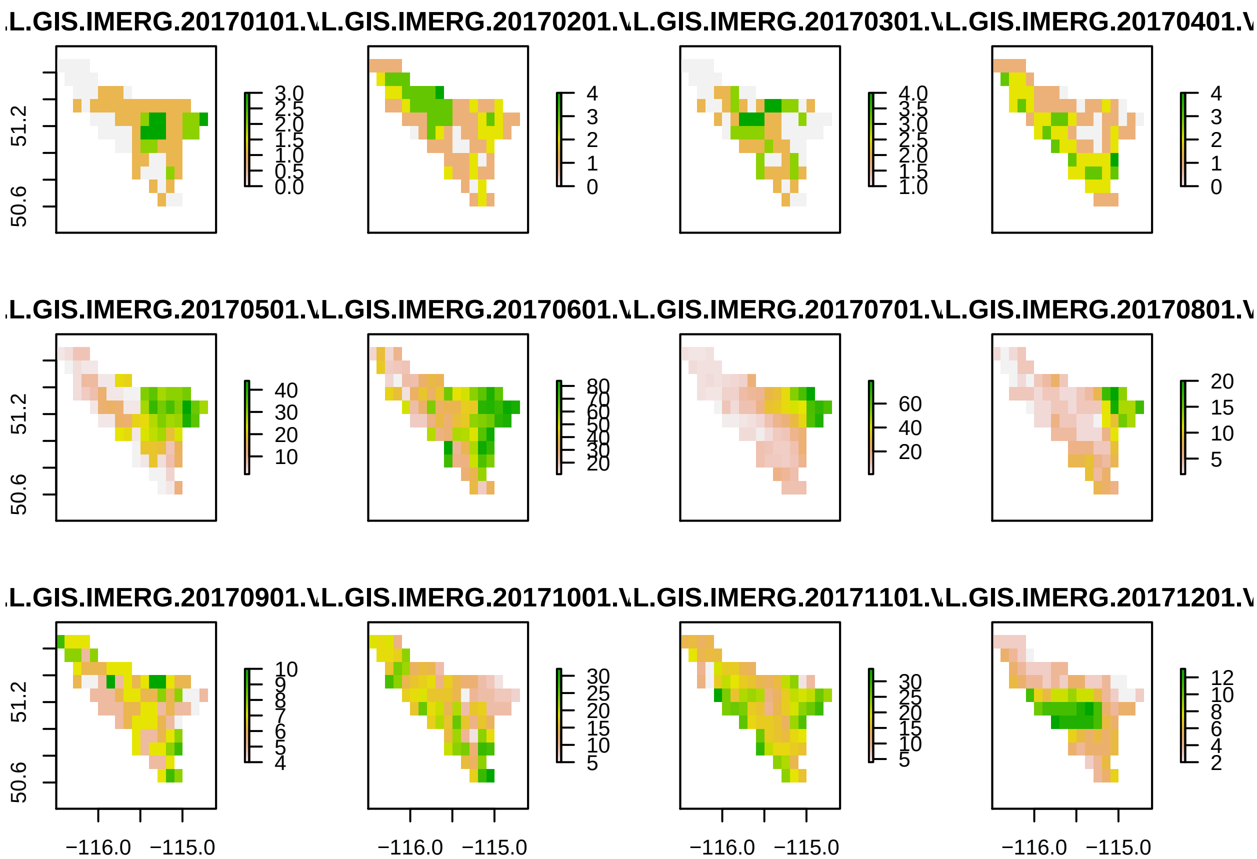
Loading and ploting monthly (2017) precipitation from GPM/IMERG

```
GPM_list <- list.files(path = getwd(), pattern = "3B-MO-L.GIS.IMERG.*.tif$")

GPM_2017 <- stack(GPM_list)

GPM_BowBasin <- crop(GPM_2017, AOI)

fun <- function() {
  plot(BowRiver_atCochrane, add = T, bg = "transparent")
}

plot(GPM_BowBasin, addfun = fun)
```

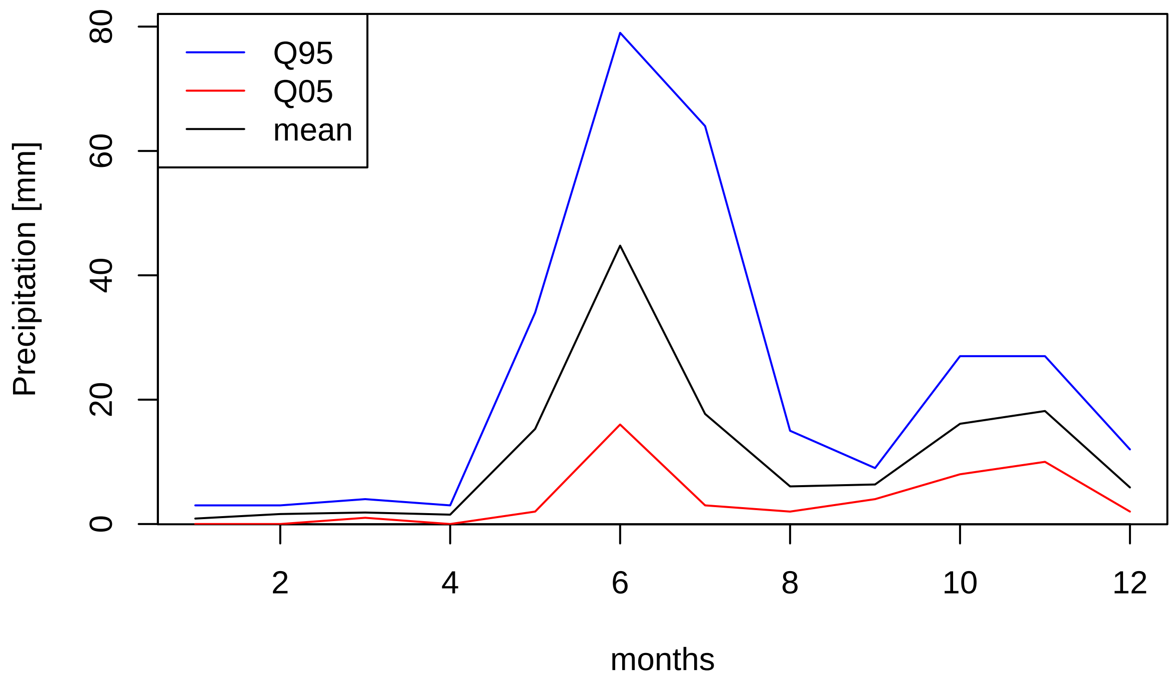


Extracting spatial statistics from a basin

```
GPM_BowBasin_mask <- mask(GPM_BowBasin, BowRiver_atCochrane)

plot(GPM_BowBasin_mask)
```

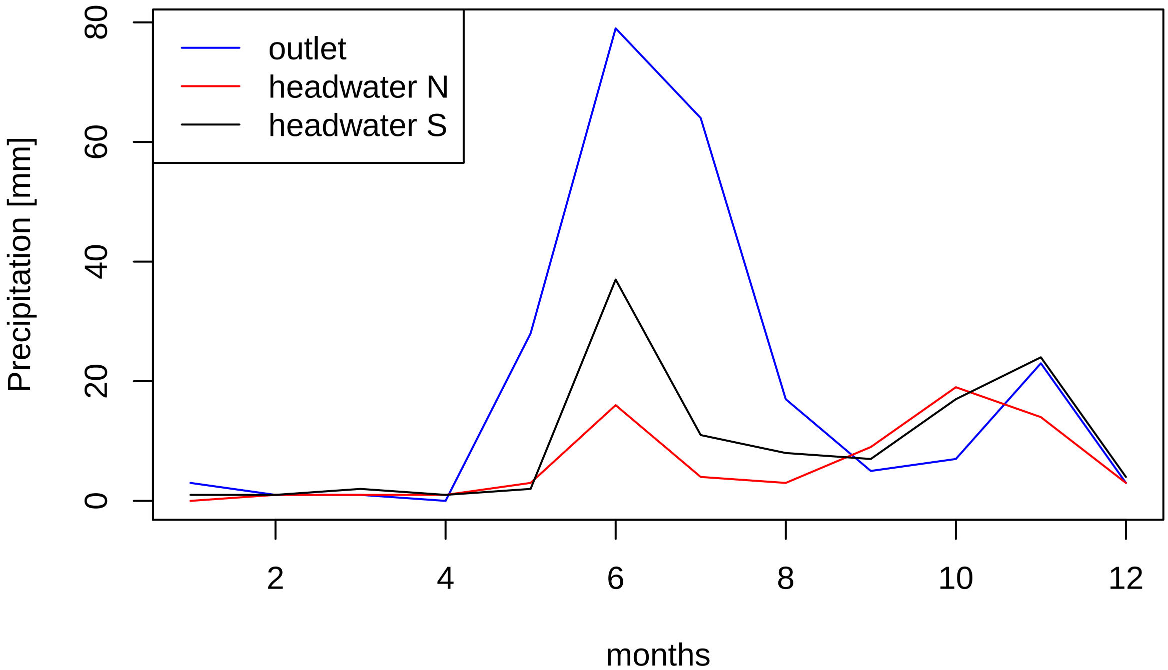



```
GPM_mean <- cellStats(GPM_BowBasin_mask, stat = "mean")

GPM_Q95 <- quantile(GPM_BowBasin_mask, 0.95)
GPM_Q05 <- quantile(GPM_BowBasin_mask, 0.05)

months <- 1:12

plot(months, GPM_Q95, type = "l", col = "blue", ylab = "Precipitation [mm]")
lines(months, GPM_Q05, col = "red")
lines(months, GPM_mean, col = "black")
legend("topleft", legend = c("Q95", "Q05", "mean"),
       col = c("blue", "red", "black"), lty = c(1,1,1))
```



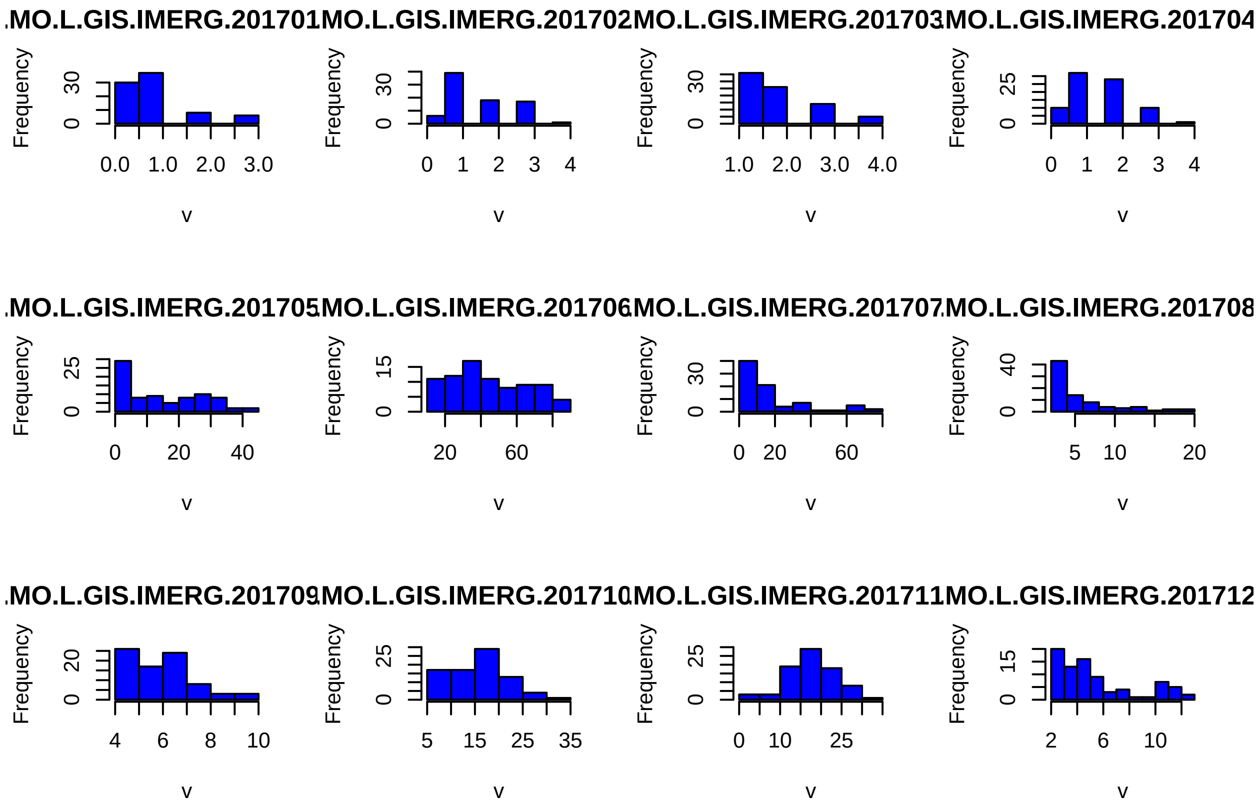
Extracting values from XY points from image time series

```
X <- c(-114.75, -116.45, -115.24)
Y <- c(51.24, 51.64, 50.64)

points <- data.frame(X, Y)

spt_points <- SpatialPoints(points)

points_precip <- extract(GPM_BowBasin_mask, spt_points)

plot(months, points_precip[1,], type = "l", col = "blue", ylab = "Precipitation [mm]")
lines(months, points_precip[2,], col = "red")
lines(months, points_precip[3,], col = "black")
legend("topleft", legend = c("outlet", "headwater N", "headwater S"),
       col = c("blue", "red", "black"), lty = c(1,1,1))
```



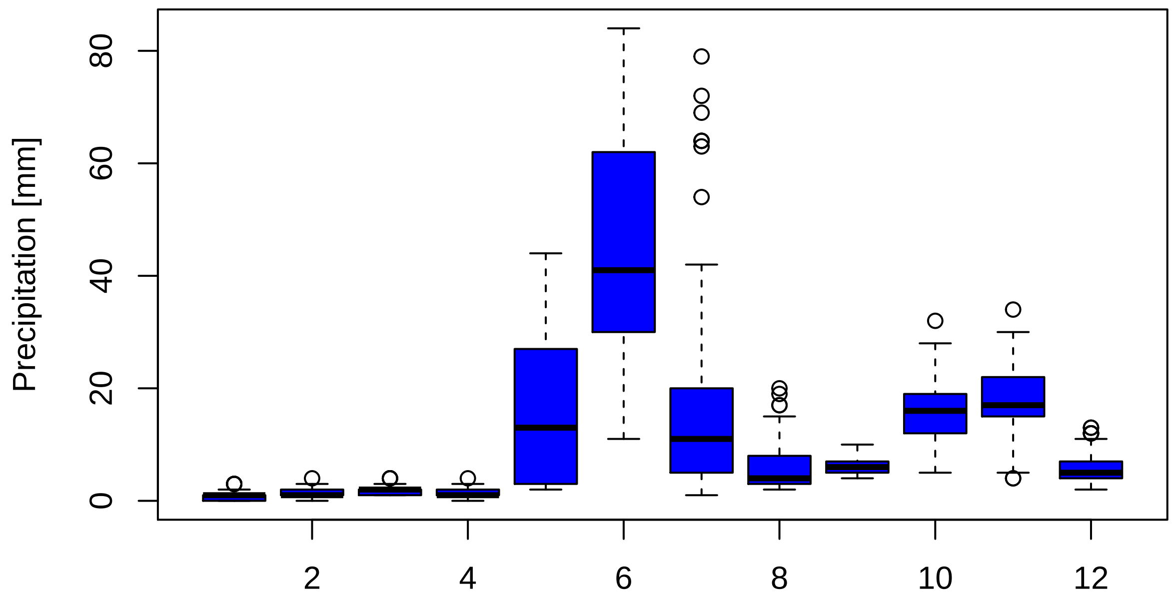
Spatial histogram

```
hist(GPM_BowBasin_mask, col = "blue")
```



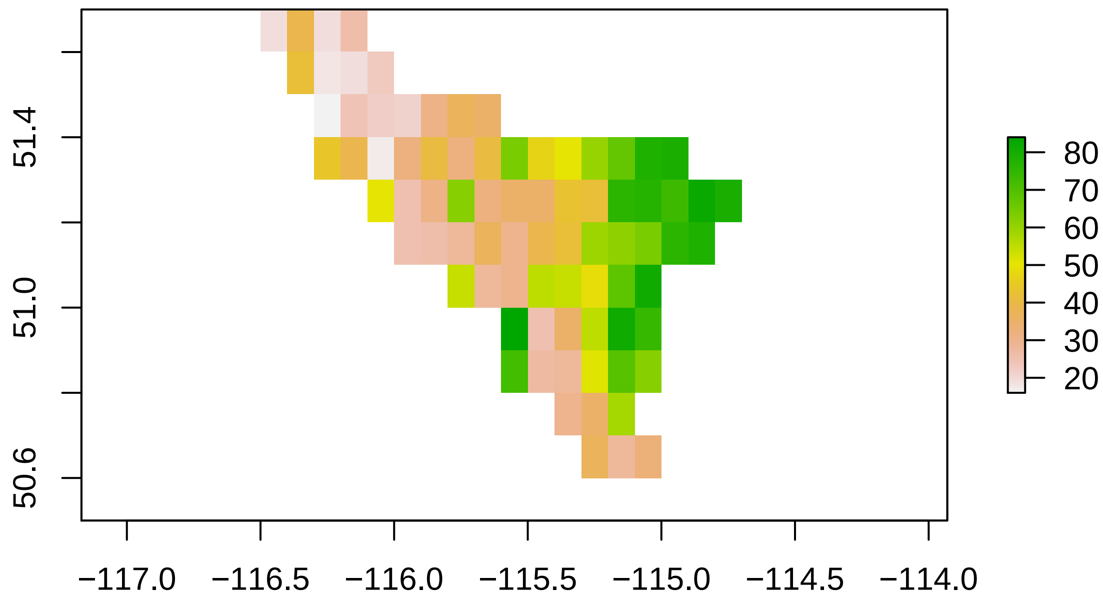
Spatial boxplots

```
boxplot(GPM_BowBasin_mask, col = "blue", ylab = "Precipitation [mm]", xaxt = "n")
axis(months)
```



Temporal statistics

```
annual_max <- max(GPM_BowBasin_mask)

plot(annual_max)
```



Principal Component Analysis

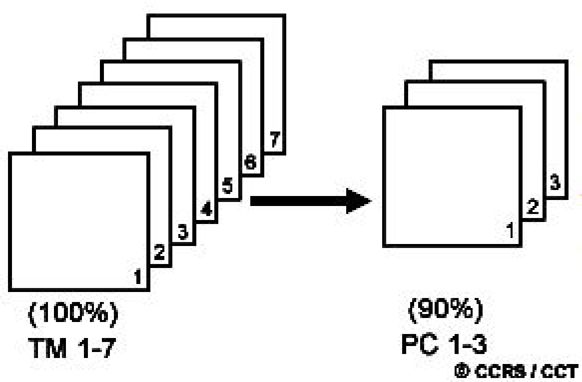
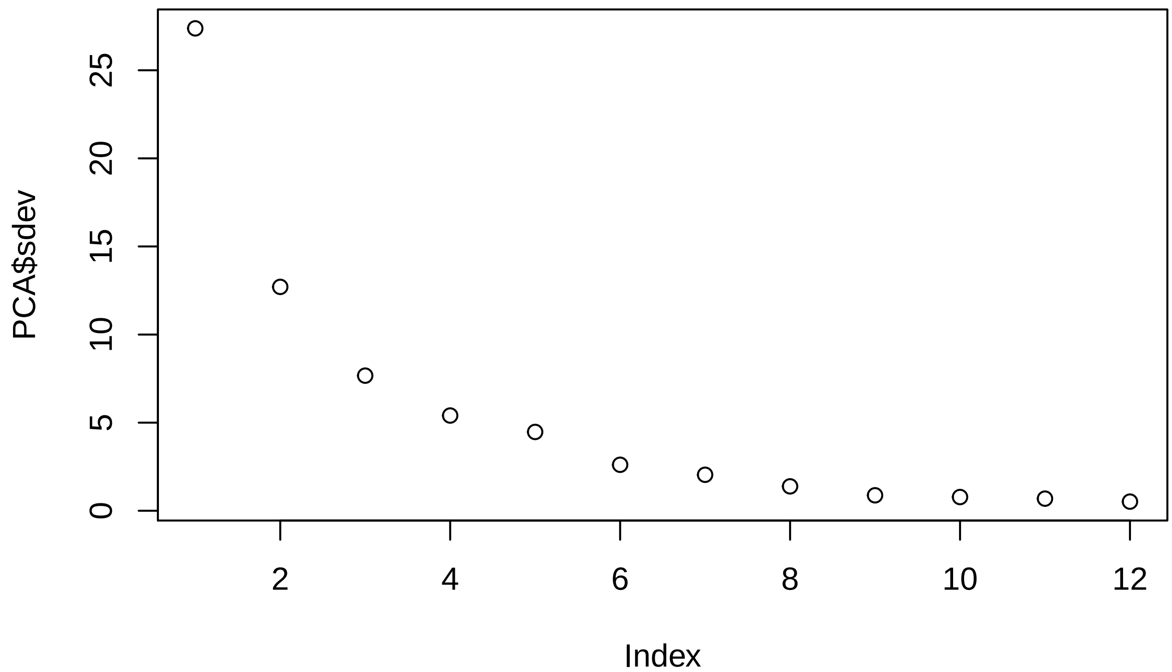


Figure 4: Reducing Dimensionality (CCRS, 2015)

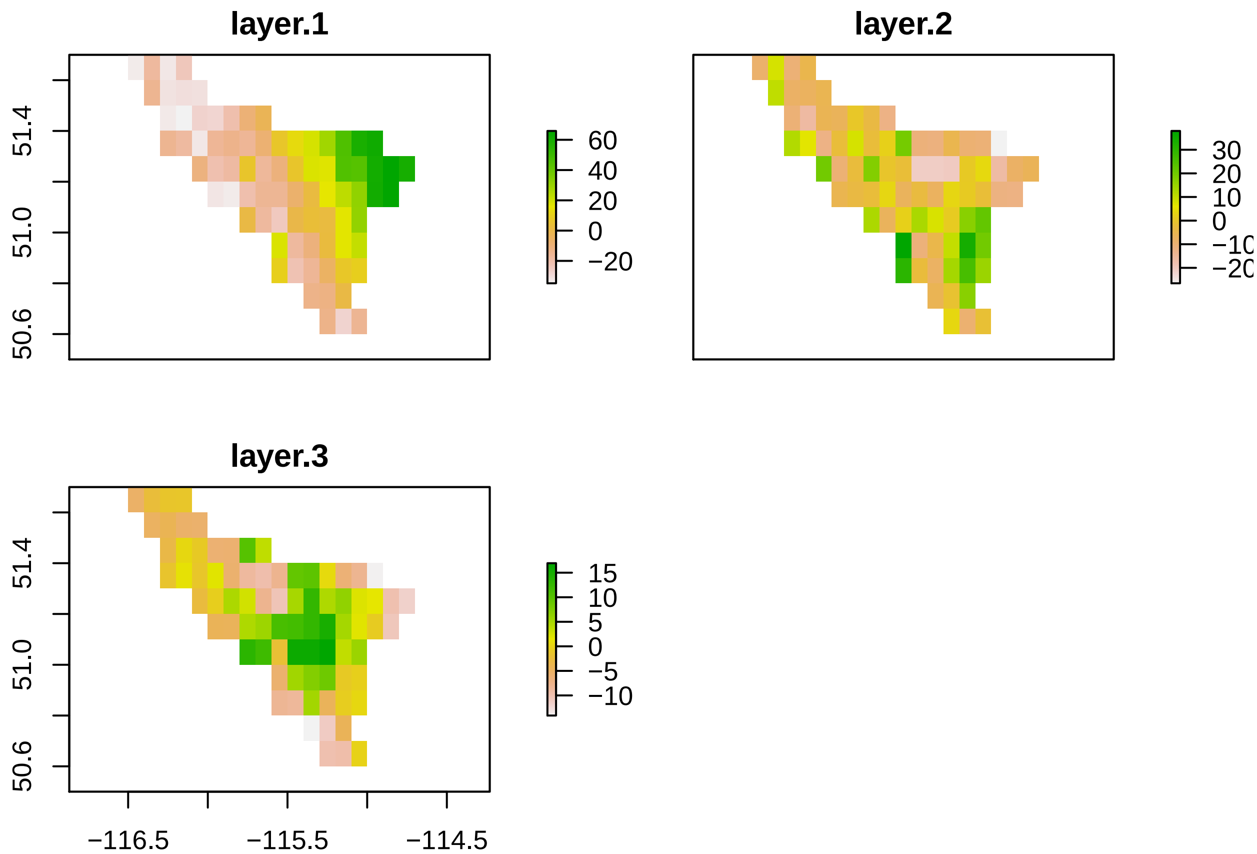
```
sr <- sampleRandom(GPM_BowBasin_mask, 100)
PCA <- prcomp(sr)

plot(PCA$sdev)
```




```
print(sum(PCA$sdev[1:3]))
```

```
## [1] 47.75133
```

```
x <- predict(GPM_BowBasin_mask, PCA, index=1:3)
plot(x)
```



layer.1
layer.2
layer.3
51.4
51.0
50.6
−116.5
−115.5
−114.5
60
40
20
0
−20
30
20
10
0
−10
−20
15
10
5
0
−5
−10