This report covers the code for the summary table and summary section from the AIM Annual report. For the purpose, methods and example reporting see the “Annual report code purpose and methods” report. A brief summary of methods will be provided in this report.

For a detailed description of the code below see the “Annual report code..”. This code must be run prior to any of the code below. If any section of code is highlighted more detail about it will be given following the end of the code

```
#packages installed= data.table, dplyr, formattable, tidyr using function
install.packages("") for example install.packages ("tidyr")
library(tidyr)
library(ggplot2)
library(data.table)
library(dplyr)
library(formattable)
library(qwraps2)


#Actual.Eco.Site is a column I added to each TerrADat csv to assign the ecological site
#CSVs have to be saved as CSV with the names as follows: plots = "Allyears_plots" , soil
horizon = "Allyears_soilhorizons"
#query results = "Allyears_query", plant specis = "PSPPALL" , species richness =
"Allyears_species_rich"
#LPI detail = "LPI_all"

#Plotdata has the actual.eco.site assigned, the "match" code below is adding a column that
will assign actual.eco.site to every row
##using the Plotdata csv
Plotdata<-read.csv("~/Allyears_plots.csv")
soildata<-read.csv("~/Allyears_soilhorizons.csv")
soildata$Actual.Eco.Site<-
Plotdata$Actual.Eco.Site[match(soildata$PrimaryKey,Plotdata$PrimaryKey)]
AIMdata<-read.csv("~/Allyears_query.csv")
AIMdata$Actual.Eco.Site<-
Plotdata$Actual.Eco.Site[match(AIMdata$Primary.Key,Plotdata$PrimaryKey)]
PSPSdata<-read.csv("~/PSPPALL.csv")
PSPSdata$Actual.Eco.Site<-
Plotdata$Actual.Eco.Site[match(PSPSdata$PrimaryKey,Plotdata$PrimaryKey)]
SPdata<-read.csv("~/Allyears_species_rich.csv")
SPdata$Actual.Eco.Site<-
Plotdata$Actual.Eco.Site[match(SPdata$PrimaryKey,Plotdata$PrimaryKey)]
ST<-read.csv("~/LPI_all.csv")
#when downloading LPI detail from TerrADat online you will have to label the PrimaryKey
column PrimaryKey, you can use the View()
```

```
##command to ensure you're matching with the correct column
ST<-ST %>%
  select(HeightWoody,HeightHerbaceous,SpeciesWoody,SpeciesHerbaceous,PrimaryKey)
STN<-na.omit(ST)
STN$Actual.Eco.Site<-Plotdata$Actual.Eco.Site[match(STN$PrimaryKey,Plotdata$PrimaryKey)]

#replace with abbreviation of your ecosite. abbrev of eco sites in quotes in target()
#which can be done using ctrl + f , replace all
```

Noxious and Nonnox. Species

```
#native vs nonnative
target<-c("BH","LB","LOA","MO","OTH","ROF","SAL","SAN","FORE")
AIMdata1<-AIMdata%>%
  select(Noxious.Cover.Pct.Any.Hit,NonNoxious.Plant.Cover.Pct.Any.Hit,Actual.Eco.Site) %>%
  filter(Actual.Eco.Site %in% target)



AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site)

Covers<-c("Nonnox.", "Noxious")
NN<-ggplot(AIMdata2,aes(x=covertype,y=covervalue, col=covertype))+geom_point(alpha=0.4)
NN2<-NN+geom_boxplot()
NN3<-NN2+ggtitle("Noxious and Nonnoxious Species Comparison by Strata")+labs(x="Cover
Type",y="Average Percent Cover",colour="Cover Type")
NN4<-NN3+scale_x_discrete(labels=
Covers)+scale_color_manual(values=c("seagreen2","darkred"),labels = c("Nonnoxious",
"Noxious"))+theme(axis.text.x=element_text(colour="gray20"))
NN4+facet_wrap(~Actual.Eco.Site,
ncol=3)+theme(axis.text.x=element_text(colour="gray20",angle = 45, hjust=1))
```

TerrADat Query Results csv was used in noxious and nonnoxious species calculations. Specifically columns Noxious.Cover.Pct.Any.Hit and NonNoxious.Plant.Cover.Pct.Any.Hit were used. TerrADat uses LPI results and CDA noxious species lists A, B, C and watch for the calculations. %>% indicates that you are continuing the code for that step. Target is the Actual.Eco.Sites you are interested in viewing. This is useful if your csv has allotments, strata, intensifications etc. that you aren't currently interested in. To do this, set target and use the filter(%in%) function in the following lines. Ggplot() is commonly used throughout the entire code. This sets a base scatterplot which can be built on. You can do all of the following: changing the style of graph with geom_(), changing the name with ggtitle(), changing axes labels with labs(), changing colors with scale_color_manual(), changing text with theme() or changing the plots orientation with facet_wrap().

Perennial and Annual Grass

```
#perennial and annual grass
AIMdata1<-AIMdata %>%
  select(Perennial.Grass.Cover.Pct.Any.Hit,Annual.Grass.Cover.Pct.Any.Hit,Actual.Eco.Site)
%>%
  filter(Actual.Eco.Site %in% target)
AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site)

Covers<-c("Annual Grass","Perennial Grass")
G<-ggplot(AIMdata2, aes(x=covertype,y=covervalue,col=covertype))
G2<-G+geom_boxplot()
G3<-G2+ggtitle(" Percent Cover of Annual and Perennial
Grass")+labs(x="Duration",y="Average Percent
Cover",colour="Duration")+scale_x_discrete(labels= Covers)
G3+facet_wrap(.~Actual.Eco.Site,
ncol=3)+theme(axis.text.x=element_text(colour="gray20",angle = 45, hjust=1))+
scale_color_manual(values=c("paleturquoise3", "rosybrown3"),labels = c("Perennial",
"Annual"))
```

TerrADat Query Results columns Perennial.Grass.Cover.Pct.Any.Hit andAnnual.Grass.Cover.Pct.Any.Hit were used in the above calculations. TerrADat uses LPI results and the classification of growth form from USDA PLANT SPECIES list.

The following steps are run before getting to the summary table section of the code. #surface cover and #calculating population size are both needed for the surface cover summary graph (later in this report)

```
#surface cover
AIMdata$shrub.subshrub<-
AIMdata$Shrub.Cover.Pct.Any.Hit+AIMdata$Noxious.SubShrub.Cover.Pct.Any.Hit+AIMdata$NonNoxi
ous.SubShrub.Cover.Pct.Any.Hit
AIMdata$succulent<-
AIMdata$Noxious.Succulent.Cover.Pct.Any.Hit+AIMdata$NonNoxious.Succulent.Cover.Pct.Any.Hit
AIMdata$tree<-
AIMdata$Noxious.Tree.Cover.Pct.Any.Hit+AIMdata$NonNoxious.Tree.Cover.Pct.Any.Hit
AIMdata$noxiousforb<-
AIMdata$Noxious.Annual.Forb.Cover.Pct.Any.Hit+AIMdata$Noxious.Perennial.Forb.Cover.Pct.Any
.Hit
```

```
AIMdata$nonnoxiousforb<-
AIMdata$NonNoxious.Annual.Forb.Cover.Pct.Any.Hit+AIMdata$NonNoxious.Perennial.Forb.Cover.P
ct.Any.Hit
```

The above code gives some of the columns used from the TerrADat Query Results for calculations. AIMdata being the TerrADat Query Results and $ indicates the column name.

```
#calculating Population Size, will be used throughout the entire script
BH<-subset(AIMdata,Actual.Eco.Site=="BH")

BHN<-count(BH,Primary.Key)
BHN2<-sum(BHN$n)
```

Subset() function allows for the selecting of only one Actual.Eco.Site group which you are interested in. Count() gives the total number of observations (primary key is unique even when the same plot is sampled in different years). Sum() adds the values.

```
#calculating surface cover

AIMdata1<-AIMdata %>%

select(Bare.Soil.Pct,Forb.Cover.Pct.Any.Hit,Grass.Cover.Pct.Any.Hit,Total.Litter.Cover.Pct
.First.Hit,Rock.Cover.Pct.First.Hit,shrub.subshrub,succulent,tree,Foliar.Cover.Pct,Actual.
Eco.Site)

AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site)
Covers<-c("Bare Soil", "Forb","Grass", "Litter",
"Rock","Shrub","Succulent","Tree","Foliar")
AIMdata2$covertype <- factor(AIMdata2$covertype, levels =
c("Bare.Soil.Pct","Forb.Cover.Pct.Any.Hit","Grass.Cover.Pct.Any.Hit","Total.Litter.Cover.P
ct.First.Hit","Rock.Cover.Pct.First.Hit","shrub.subshrub","succulent","tree","Foliar.Cover
.Pct","Actual.Eco.Site"))

BH<-subset(AIMdata2,Actual.Eco.Site=="BH")
```

Select() function allows us to choose only the columns we're interested in. The select step provides all columns from TerrADat Query Results that were used in the calculations.

```
BHpop<-paste("BH Surface Cover   n=",BHN2)
```

At this point the dominant species code needs to be run. See the Annual report…for detailed description of this code.

```
#gap barchart
Gapdata<-AIMdata %>%

select(Gap.Cover.25.to.50.Pct,Gap.Cover.51.to.100.Pct,Gap.Cover.101.to.200.Pct,Gap.Cover.20
0.Plus.Pct,Actual.Eco.Site)
Gapdata$CanopyCover<-100-
(AIMdata$Gap.Cover.25.to.50.Pct+AIMdata$Gap.Cover.51.to.100.Pct+AIMdata$Gap.Cover.101.to.20
0.Pct+AIMdata$Gap.Cover.200.Plus.Pct)
Gapdata2<-gather(Gapdata,Gapsize,Percentofplot,-Actual.Eco.Site)
Gapdata2$Gapsize <- factor(Gapdata2$Gapsize, levels =
c("CanopyCover","Gap.Cover.25.to.50.Pct", "Gap.Cover.51.to.100.Pct",
"Gap.Cover.101.to.200.Pct","Gap.Cover.200.Plus.Pct"))

BHG2<-subset(Gapdata2,Actual.Eco.Site=="BH")

BHpop<-paste("BH Proportion Canopy Gap and Cover   n=",BHN2)
```

The Gaps category is calculated using the TerrADat gap columns (see select step for specific columns). Canopy cover is calculated by subtracting the gaps value from 100 since canopy cover and gap should total 100%.

#species richness

```
BHSR<-subset(SPdata,Actual.Eco.Site=="BH")
```

```
BH<-subset(AIMdata,Actual.Eco.Site=="BH")
```

Summary Table

The following is to calculate shrub and tree heights.

```
#for the below code replace "insert dom ss..." with dominant species names for the correct
functional group. if none = "NA" . If more than one
#for a function al group the name has to be added to Indicator as "species name",
#then added to average using round(mean(BHA$Species=="insert stratas dominant grass
species, if none = NA"), digits=2),
#added to min using round(min(BHA$Species=="insert stratas dominant grass species, if none
= NA"), digits=2),
#added to max and St.Dev using same code replacing the "min" with "max" and "sd"
#calculate shrub and tree height data using LPI detail

PS2<-PSPSdata[!duplicated(PSPSdata$Species),]
```

```
STN$GrowthHabitsub<-PS2$GrowthHabitSub[match(STN$SpeciesWoody,PS2$Species)]

BHST<-subset(STN,Actual.Eco.Site=="BH")
```

PS2 uses the species csv from TerrADat, which gives the growth form and habit of all species recorded for each plot. STN is created in the beginning of the code and is the TerrADat LPI detail csv. This code used the match() function to match the species from PS2 (species csv) to the species in STN (LPI detail) to create a growth habit column in STN.

```
BHT<-BHST %>%
  filter(GrowthHabitsub=="Tree")
BHS<-BHST %>%
  filter(GrowthHabitsub=="Shrub")
```

The filter() function allows us to select only the growth habits we are interested in. The calculations for mean(), max() and min() will be completed in the following steps in the table. However, this can be viewed now using summary(BHT). You can also add this column to the Query Results using the match() function and primarykey columns.

Putting it all together:

```
#BH
#replace your dominant species codes (determined in a previous step) in the below code
Indicator<-c("Surface Indicator (%)","Rock Fragment", "Litter","Bare Soil",
             "Foliar Cover (%)","Shrub/Sub-shrub","Grass","Succulent",
             "Tree","Noxious Forb/Herb","Nonnoxious Forb/Herb","Vegetation Height (cm)",
             "Grass","Forb/Herb","Tree", "Shrub",
             "Dominant Graminoid Cover (%)","BOGR2","HECO26",
             "Dominant Herb. Cover (%)", "NA",
             "Dominant Woody Cover (%)","KRLA2","PIED",
             "Canopy Gaps (%)", "Gaps 25-50 cm", "Gaps 51-100 cm",
             "Gaps 101-200 cm","Gaps >200 cm", "Canopy Cover (%)",
             "Total Canopy Cover","Species Richness","No. Species per Plot")

#replace with your dominant species plant codes. If only one plant code remove everything after digits=2) except the parantheses
#for example the code below would become domgrassme<-c(round(mean(Species1$AH_SpeciesCover, na.rm=TRUE), digits=2))
#if there is no dominant species for the growth form list "NA" as seen below in domherme<-"NA"
Species1<- subset(BH8,Species=="BOGR2")
Species2<-subset(BH8,Species=="HECO26")
Species3<-subset(BH8,Species=="KRLA2")
Species4<-subset(BH8,Species=="PIED")
```

<-replace the Species== “” with your species name that was determined during the dominant species code

<-you are inserting your dominant species into the correct column on the table. Domgrass are dominant grass species, domherb are dominant herbaceous species and domwood are dominant woody species. If there are no species in that category delete any text and change to “NA” as in the domherb category. If there are species in the category use round(neededfunction(Species#$AH_SpeciesCover, na.rm=TRUE),digits=2). For multiple species use c() and separate with a comma as in the domgrass category. Code is separated by function (me=mean, ma=max, mi=min, etc.)

```
domgrassme<-c(round(mean(Species1$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(mean(Species2$AH_SpeciesCover, na.rm=TRUE),digits=2))
domherbme<-"NA"
domwoodme<-c(round(mean(Species3$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(mean(Species4$AH_SpeciesCover, na.rm=TRUE), digits=2))

domgrassma<-c(round(max(Species1$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(max(Species2$AH_SpeciesCover, na.rm=TRUE), digits=2))
domherbma<-"NA"
domwoodma<-c(round(max(Species3$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(max(Species4$AH_SpeciesCover, na.rm=TRUE), digits=2))

domgrassmi<-c(round(min(Species1$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(min(Species2$AH_SpeciesCover, na.rm=TRUE), digits=2))
domherbmi<-"NA"
domwoodmi<-c(round(min(Species3$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(min(Species4$AH_SpeciesCover, na.rm=TRUE), digits=2))

domgrasssd<-c(round(sd(Species1$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(sd(Species2$AH_SpeciesCover, na.rm=TRUE), digits=2))
domherbsd<-"NA"
domwoodsd<-c(round(sd(Species3$AH_SpeciesCover, na.rm=TRUE),
digits=2),round(sd(Species4$AH_SpeciesCover, na.rm=TRUE), digits=2))

domgrassCI<-c(round((1.282*(sd(Species1$AH_SpeciesCover, na.rm=TRUE)/BHN2)),digits = 2) ,
round((1.282*(sd(Species2$AH_SpeciesCover, na.rm=TRUE)/BHN2)),digits=2))
domherbCI<-"NA"
domwoodCI<-c(round((1.282*(sd(Species3$AH_SpeciesCover,
na.rm=TRUE)/BHN2)),digits=2),round((1.282*(sd(Species4$AH_SpeciesCover,
na.rm=TRUE)/BHN2)),digits=2))
```

<-no editing needed after this point

```
Mean<-c("",round(mean(BH$Rock.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(mean(BH$Total.Litter.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(mean(BH$Bare.Soil.Pct, na.rm=TRUE), digits=2),
        "",round(mean(BH$shrub.subshrub, na.rm=TRUE), digits=2),
round(mean(BH$Grass.Cover.Pct.Any.Hit, na.rm=TRUE),digits=2),round(mean(BH$succulent,
na.rm=TRUE),digits=2),
        round(mean(BH$tree, na.rm=TRUE),digits=2),round(mean(BH$noxiousforb,
na.rm=TRUE),digits=2),round(mean(BH$nonnoxiousforb, na.rm=TRUE),digits=2),"",
        round(mean(BH$Average.Grass.Height.cm,
na.rm=TRUE),digits=2),round(mean(BH$Average.Forb.Height.cm, na.rm=TRUE),
digits=2),round(mean(BHT$HeightWoody, na.rm=TRUE), digits=2),round(mean(BHS$HeightWoody,
na.rm=TRUE), digits=2),"",
```

```r
        domgrassme,"",domherbme,"",
        domwoodme,"",round(mean(BHG$Gap.Cover.25.to.50.Pct, na.rm=TRUE), digits=2),
        round(mean(BHG$Gap.Cover.51.to.100.Pct, na.rm=TRUE),
digits=2),round(mean(BHG$Gap.Cover.101.to.200.Pct, na.rm=TRUE), digits=2),
        round(mean(BHG$Gap.Cover.200.Plus.Pct, na.rm=TRUE),
digits=2),"",round(mean(BHG$CanopyCover, na.rm=TRUE),
digits=2),"",round(mean(BHSR$SpeciesCount, na.rm=TRUE), digits=2))

Minimum<-c("",round(min(BH$Rock.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(min(BH$Total.Litter.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(min(BH$Bare.Soil.Pct, na.rm=TRUE), digits=2),
           "",round(min(BH$shrub.subshrub, na.rm=TRUE), digits=2),
round(min(BH$Grass.Cover.Pct.Any.Hit, na.rm=TRUE),digits=2),round(min(BH$succulent,
na.rm=TRUE),digits=2),
           round(min(BH$tree, na.rm=TRUE),digits=2),round(min(BH$noxiousforb,
na.rm=TRUE),digits=2),round(min(BH$nonnoxiousforb, na.rm=TRUE),digits=2),"",
           round(min(BH$Average.Grass.Height.cm,
na.rm=TRUE),digits=2),round(min(BH$Average.Forb.Height.cm, na.rm=TRUE),
digits=2),round(min(BHT$HeightWoody, na.rm=TRUE), digits=2),round(min(BHS$HeightWoody,
na.rm=TRUE), digits=2),"",
           domgrassmi,"",domherbmi,"",
           domwoodmi,"",round(min(BHG$Gap.Cover.25.to.50.Pct, na.rm=TRUE), digits=2),
           round(min(BHG$Gap.Cover.51.to.100.Pct, na.rm=TRUE),
digits=2),round(min(BHG$Gap.Cover.101.to.200.Pct, na.rm=TRUE), digits=2),
           round(min(BHG$Gap.Cover.200.Plus.Pct, na.rm=TRUE),
digits=2),"",round(min(BHG$CanopyCover, na.rm=TRUE),
digits=2),"",round(min(BHSR$SpeciesCount, na.rm=TRUE), digits=2))

Maximum<-c("",round(max(BH$Rock.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(max(BH$Total.Litter.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(max(BH$Bare.Soil.Pct, na.rm=TRUE), digits=2),
           "",round(max(BH$shrub.subshrub, na.rm=TRUE), digits=2),
round(max(BH$Grass.Cover.Pct.Any.Hit, na.rm=TRUE),digits=2),round(max(BH$succulent,
na.rm=TRUE),digits=2),
           round(max(BH$tree, na.rm=TRUE),digits=2),round(max(BH$noxiousforb,
na.rm=TRUE),digits=2),round(max(BH$nonnoxiousforb, na.rm=TRUE),digits=2),"",
           round(max(BH$Average.Grass.Height.cm,
na.rm=TRUE),digits=2),round(max(BH$Average.Forb.Height.cm, na.rm=TRUE),
digits=2),round(max(BHT$HeightWoody, na.rm=TRUE), digits=2),round(max(BHS$HeightWoody,
na.rm=TRUE), digits=2),"",
           domgrassma,"",domherbma,"",
           domwoodma,"",round(max(BHG$Gap.Cover.25.to.50.Pct, na.rm=TRUE), digits=2),
           round(max(BHG$Gap.Cover.51.to.100.Pct, na.rm=TRUE),
digits=2),round(max(BHG$Gap.Cover.101.to.200.Pct, na.rm=TRUE), digits=2),
           round(max(BHG$Gap.Cover.200.Plus.Pct, na.rm=TRUE),
digits=2),"",round(max(BHG$CanopyCover, na.rm=TRUE),
digits=2),"",round(max(BHSR$SpeciesCount, na.rm=TRUE), digits=2))

St.Dev.<-c("",round(sd(BH$Rock.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(sd(BH$Total.Litter.Cover.Pct.First.Hit, na.rm=TRUE),
digits=2),round(sd(BH$Bare.Soil.Pct, na.rm=TRUE), digits=2),
           "",round(sd(BH$shrub.subshrub, na.rm=TRUE), digits=2),
round(sd(BH$Grass.Cover.Pct.Any.Hit, na.rm=TRUE),digits=2),round(sd(BH$succulent,
na.rm=TRUE),digits=2),
           round(sd(BH$tree, na.rm=TRUE),digits=2),round(sd(BH$noxiousforb,
na.rm=TRUE),digits=2),round(sd(BH$nonnoxiousforb, na.rm=TRUE),digits=2),"",
           round(sd(BH$Average.Grass.Height.cm,
na.rm=TRUE),digits=2),round(sd(BH$Average.Forb.Height.cm, na.rm=TRUE),
digits=2),round(sd(BHT$HeightWoody, na.rm=TRUE), digits=2),round(sd(BHS$HeightWoody,
na.rm=TRUE), digits=2),"",
           domgrasssd,"",domherbsd,"",
           domwoodsd,"",round(sd(BHG$Gap.Cover.25.to.50.Pct, na.rm=TRUE), digits=2),
           round(sd(BHG$Gap.Cover.51.to.100.Pct, na.rm=TRUE),
digits=2),round(sd(BHG$Gap.Cover.101.to.200.Pct, na.rm=TRUE), digits=2),
           round(sd(BHG$Gap.Cover.200.Plus.Pct, na.rm=TRUE),
digits=2),"",round(sd(BHG$CanopyCover, na.rm=TRUE),
digits=2),"",round(sd(BHSR$SpeciesCount, na.rm=TRUE), digits=2))

ME<-c("", round((1.282*sd(BH$Rock.Cover.Pct.First.Hit, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$Total.Litter.Cover.Pct.First.Hit, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$Bare.Soil.Pct, na.rm=TRUE)/BHN2),digits=2),
      "", round((1.282*sd(BH$shrub.subshrub, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$Grass.Cover.Pct.Any.Hit, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$succulent, na.rm=TRUE)/BHN2),digits=2),
      round((1.282*sd(BH$tree, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$noxiousforb, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$nonnoxiousforb, na.rm=TRUE)/BHN2),digits=2),"",
      round((1.282*sd(BH$Average.Grass.Height.cm, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BH$Average.Forb.Height.cm, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BHT$HeightWoody, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BHS$HeightWoody, na.rm=TRUE)/BHN2),digits=2),"",
      domgrassCI,"",domherbCI,"",
      domwoodCI,"", round((1.282*sd(BHG$Gap.Cover.25.to.50.Pct,
na.rm=TRUE)/BHN2),digits=2),
      round((1.282*sd(BHG$Gap.Cover.51.to.100.Pct, na.rm=TRUE)/BHN2),digits=2),
round((1.282*sd(BHG$Gap.Cover.101.to.200.Pct, na.rm=TRUE)/BHN2),digits=2),
      round((1.282*sd(BHG$Gap.Cover.200.Plus.Pct, na.rm=TRUE)/BHN2),digits=2),"",
round((1.282*sd(BHG$CanopyCover, na.rm=TRUE)/BHN2),digits=2),"",
round((1.282*sd(BHSR$SpeciesCount, na.rm=TRUE)/BHN2),digits=2))


BHdataframe<-data.frame(Indicator,Mean,Minimum,Maximum,St.Dev.,ME)
BHdataframe<-BHdataframe %>% mutate_if(is.factor, list(~na_if(., Inf))) %>%
  mutate_if(is.factor, list(~na_if(., -Inf)))%>%
  mutate_if(is.factor, list(~na_if(., "NaN")))
target<-c("Surface Indicator (%)","Foliar Cover (%)","Vegetation Height (cm)","Dominant
Graminoid Cover (%)",
          "Dominant Herb. Cover (%)","Dominant Woody Cover (%)","Canopy Gaps (%)","Canopy
Cover (%)",
          "Species Richness")
blank_bold<-formatter("span",
                      style = x ~ style("font-weight" = ifelse(x %in% target, "bold", NA)))
BHsummtab<-formattable(BHdataframe, list(
  Indicator = blank_bold))
BHsummtab
#run if table is too large to view in export, this will send it to excel
write.csv(BHsummtab,"BHsummtab.csv")
```

These steps used mean(), max() and min() functions. Formattable() is used to view tables and formatter() allows us to change the formatting of those tables. **All data is from TerrADat. For the shrub and tree data we simply used TerrADat species data with TerrADat LPI detail to determine the values of a specific growth form.** Keep in mind litter and rock cover in TerrADat are only percent first hit and litter is NOT between canopy. Code for determining litter cover between canopy gaps will be available on the Github 05/2020 at the latest.

Surface Cover Graph

```
AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site) %>%
  filter(Actual.Eco.Site %in% target)

Covers<-c("Bare Soil", "Forb","Grass", "Litter",
"Rock","Shrub","Succulent","Tree","Foliar")
AIMdata2$covertype <- factor(AIMdata2$covertype, levels =
c("Bare.Soil.Pct","Forb.Cover.Pct.Any.Hit","Grass.Cover.Pct.Any.Hit","Total.Litter.Cover.Pc
t.First.Hit","Rock.Cover.Pct.First.Hit","shrub.subshrub","succulent","tree","Foliar.Cover.P
ct","Actual.Eco.Site"))
```

```
SC<-ggplot(AIMdata2,aes(x=factor(covertype),y=covervalue,col=covertype))+geom_point(alpha=0.4)
SC2<-SC+geom_boxplot()+ggtitle("Surface and Vegetation Cover by Strata")+labs(x="Cover Type",y="Average Percent Cover",colour="Cover Type")
SC3<-SC2+scale_x_discrete(labels= Covers)+scale_color_hue(labels = c("Bare Soil", "Forb","Grass", "Litter","Rock", "Shrub/subshrub","Succulent","Tree", "Foliar"))+theme(axis.text.x=element_text(colour="gray20"))
SC3+facet_wrap(~Actual.Eco.Site, ncol=3)+theme(axis.text.x=element_text(colour="gray20",angle = 45, hjust=1))
```

Gather() converts our dataframe from a wide data frame to a long dataframe. Factor ensures the data (boxplots) will be displayed in the correct order. In other words, we need to make sure the correct boxplot is aligning with the names we are assigning on the x-axis. Ggplot() is the plotting function used throughout all of the codes. Initially it creates a scatterplot of points, but we continue to build off and modify the plot with functions including geom_boxplot, scale_x_discrete (this is used to change the labels but can change size of text, font etc for the x-axis), scale_color_hue, theme(this can change text size, color and font, and the background for the entire graph) and facet_wrap (this changes the layout of graphs).

Comparing Strata

```
#bare soil cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Bare.Soil.Pct) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_baresoil=mean(Bare.Soil.Pct)) %>%
  select(Actual.Eco.Site,avg_baresoil) %>%
  arrange(desc(avg_baresoil))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_baresoil),]
formattable(AIMdata3)

#forb cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Forb.Cover.Pct.Any.Hit) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_forb=mean(Forb.Cover.Pct.Any.Hit)) %>%
  select(Actual.Eco.Site,avg_forb) %>%
  arrange(desc(avg_forb))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_forb),]
formattable(AIMdata3)
```

```
#grass cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Grass.Cover.Pct.Any.Hit) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_grass=mean(Grass.Cover.Pct.Any.Hit)) %>%
  select(Actual.Eco.Site,avg_grass) %>%
  arrange(desc(avg_grass))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_grass),]
formattable(AIMdata3)

#litter cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Total.Litter.Cover.Pct.First.Hit) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_litter=mean(Total.Litter.Cover.Pct.First.Hit)) %>%
  select(Actual.Eco.Site,avg_litter) %>%
  arrange(desc(avg_litter))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_litter),]
formattable(AIMdata3)

#rock cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Rock.Cover.Pct.First.Hit) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_rock=mean(Rock.Cover.Pct.First.Hit)) %>%
  select(Actual.Eco.Site,avg_rock) %>%
  arrange(desc(avg_rock))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_rock),]
formattable(AIMdata3)

#shrub cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,shrub.subshrub) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_shrub=mean(shrub.subshrub)) %>%
  select(Actual.Eco.Site,avg_shrub) %>%
  arrange(desc(avg_shrub))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_shrub),]
formattable(AIMdata3)

#succulent cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,succulent) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_succulent=mean(succulent)) %>%
  select(Actual.Eco.Site,avg_succulent) %>%
  arrange(desc(avg_succulent))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_succulent),]
formattable(AIMdata3)

#tree cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,tree) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_tree=mean(tree)) %>%
  select(Actual.Eco.Site,avg_tree) %>%
  arrange(desc(avg_tree))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_tree),]
formattable(AIMdata3)

#foliar cover
AIMdata3<- AIMdata1 %>%
  select(Actual.Eco.Site,Foliar.Cover.Pct) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_foliar=mean(Foliar.Cover.Pct)) %>%
  select(Actual.Eco.Site,avg_foliar) %>%
  arrange(desc(avg_foliar))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_foliar),]
formattable(AIMdata3)

#native vs nonnative
target<-c("BH","FORE","LB","LOA","MO","OTH","ROF","SAL","SAN")
AIMdata1<-AIMdata%>%
  select(Noxious.Cover.Pct.Any.Hit,NonNoxious.Plant.Cover.Pct.Any.Hit,Actual.Eco.Site) %>%
  filter(Actual.Eco.Site %in% target)

AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site)
head(AIMdata2)

Covers<-c("Nonnox.", "Noxious")
NN<-ggplot(AIMdata2,aes(x=covertype,y=covervalue, col=covertype))+geom_point(alpha=0.4)
NN2<-NN+geom_boxplot()
NN3<-NN2+ggtitle("Noxious and Nonnoxious Species Comparison by Strata")+labs(x="Cover Type",y="Average Percent Cover",colour="Cover Type")
```

```
NN4<-NN3+scale_x_discrete(labels=
Covers)+scale_color_manual(values=c("seagreen2","darkred"),labels = c("Nonnoxious",
"Noxious"))+theme(axis.text.x=element_text(colour="gray20"))
NN4+facet_wrap(~Actual.Eco.Site,
ncol=3)+theme(axis.text.x=element_text(colour="gray20",angle = 45, hjust=1))

AIMdata1<- AIMdata1 %>%
  select(Actual.Eco.Site,Noxious.Cover.Pct.Any.Hit, NonNoxious.Plant.Cover.Pct.Any.Hit) %>%
  group_by(Actual.Eco.Site) %>%
  mutate(avg_nox=mean(Noxious.Cover.Pct.Any.Hit)) %>%
  mutate(avg_nonox=mean(NonNoxious.Plant.Cover.Pct.Any.Hit))

AIMdata3<-AIMdata1 %>%
  select(Actual.Eco.Site,avg_nox) %>%
  arrange(desc(avg_nox))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_nox),]
formattable(AIMdata3)

AIMdata3<- AIMdata1 %>%
select(Actual.Eco.Site,avg_nonox) %>%
  arrange(desc(avg_nonox))
AIMdata3<-AIMdata3[!duplicated(AIMdata3$avg_nonox),]
formattable(AIMdata3)

#perennial and annual grass
AIMdata1<-AIMdata %>%
  select(Perennial.Grass.Cover.Pct.Any.Hit,Annual.Grass.Cover.Pct.Any.Hit,Actual.Eco.Site)
%>%
  filter(Actual.Eco.Site %in% target)
AIMdata2<-gather(AIMdata1,covertype,covervalue,-Actual.Eco.Site)

Covers<-c("Annual Grass","Perennial Grass")
G<-ggplot(AIMdata2, aes(x=covertype,y=covervalue,col=covertype))
G2<-G+geom_boxplot()
G3<-G2+ggtitle(" Percent Cover of Annual and Perennial Grass")+labs(x="Duration",y="Average
Percent Cover",colour="Duration")+scale_x_discrete(labels= Covers)
G3+facet_wrap(.~Actual.Eco.Site,
ncol=3)+theme(axis.text.x=element_text(colour="gray20",angle = 45, hjust=1))+
scale_color_manual(values=c("paleturquoise3", "rosybrown3"),labels = c("Perennial",
"Annual"))

#canopy gap chart
Gapdata<-AIMdata %>%
select(Gap.Cover.25.to.50.Pct,Gap.Cover.51.to.100.Pct,Gap.Cover.101.to.200.Pct,Gap.Cover.20
0.Plus.Pct,Actual.Eco.Site)
Gapdata$CanopyCover<-100-
(AIMdata$Gap.Cover.25.to.50.Pct+AIMdata$Gap.Cover.51.to.100.Pct+AIMdata$Gap.Cover.101.to.20
0.Pct+AIMdata$Gap.Cover.200.Plus.Pct)
Gapdata$Gaps<-
Gapdata$Gap.Cover.25.to.50.Pct+Gapdata$Gap.Cover.51.to.100.Pct+Gapdata$Gap.Cover.101.to.200
.Pct+Gapdata$Gap.Cover.200.Plus.Pct

Gapdatag<- Gapdata %>%
  group_by(Actual.Eco.Site) %>%
  select(Actual.Eco.Site,Gaps) %>%
  mutate(avg_gaps=mean(Gaps)) %>%
  select(Actual.Eco.Site,avg_gaps) %>%
  arrange(desc(avg_gaps))
Gapdatag<-Gapdatag[!duplicated(Gapdatag$avg_gaps),]
formattable(Gapdatag)

Gapdatac<- Gapdata %>%
  group_by(Actual.Eco.Site) %>%
  select(Actual.Eco.Site, CanopyCover) %>%
  mutate(avg_canopy=mean(CanopyCover)) %>%
  select(Actual.Eco.Site,avg_canopy) %>%
  arrange(desc(avg_canopy))
Gapdatac<-Gapdatac[!duplicated(Gapdatac$avg_canopy),]
formattable(Gapdatac)
```

! means is not. When calling on specific cells, R uses the format [row,column]. Group_by() allows you to take many subsets of the data while making a calculation. Mutate() creates a new row in the data and arrange() sorts the data least value to greatest. Changing to arrange(desc()) allows us to sort greatest to least.