Project 2 Notebook

Introduction (40 points)

- 10 points for background on the protein/gene/species of interest and where the data is sourced from

Introduction

- 10 points for specific, measurable, and clear scientific question

Scientific Question

When examining dog breeds (canis lupus familiaris), will breeds of a similar size (e.g. Cocker Spaniel, English Cocker Spaniel) have more related genes and SNP's surrounding longevity than breeds of a different size (e.g.Doberman Pinscher,Miniature Pinscher)?

Note: I only selected 6 genes most closely associated with life span (HMGA2 , IGF1 (done) , IGSF1 (too big), IRS4 (too big), LCORL (done),and SMAD2 (too big)). There are more genes involved in this, but these are the most significant.

Note: Size will be determined by the AKC. You can filter by all AKC recognized dog breeds by size. This is categorical data; if it is easier for me to work with numerical instead, I will instead use the ideal height and weight, as outlined in the official standard of each breed.

:when possible used regions by paper if regions too big used dimension of genes on ncbi

- 10 points for clear, specific, and measurable scientific hypothesis that is in the form of an if-then statement

Scientific Hypothesis

If you examine canine breeds, then breeds of a similar size (e.g.Cocker Spaniel,English Cocker Spaniel) they will have more related SNPs and or fragments of genes surrounding longevity than breeds of a different size (e.g.Doberman Pinscher,Miniature Pinscher).

- 10 points for description of what analyses were done and how the data was downloaded for the project ## Analysis Performed:

SNP's

- EDA: Scatterplots of SNP nucleotide vs size to check for visible trends before analysis
- Multiple Sequence Alignment (of SNP+border sequences), which was then visualized with msaPrettyPrint()
- Clustering of MSA results, which were visualized as Dendrograms

Gene Fragments:

- Multiple Sequence Alignment (of SNP+border sequences), which was then visualized with msaPrettyPrint() __ Clustering of MSA results, which were visualized as Dendrograms

Expression data:

- EDA: see if expression, not changes in snps is the reason behind differences

Data Sourcing

- Dog breed information (sizing): American Kennel Club LINK __ Data downloads:
- SNP and gene positionings:
- 25 points for definition of each of the packages loaded
- 5 points for correctly loading all of the packages needed

```r
#for reading in fasta files
library("BiocManager")
#for reading in excel files
library("readxl")
#for multiple sequence alignment
library("msa")
```

```
## Loading required package: Biostrings

## Loading required package: BiocGenerics

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min

## Loading required package: S4Vectors

## Loading required package: stats4

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

## Loading required package: IRanges

## Loading required package: XVector
## Loading required package: GenomeInfoDb

##
## Attaching package: 'Biostrings'

## The following object is masked from 'package:base':
##
##     strsplit

##
## Attaching package: 'msa'

## The following object is masked from 'package:BiocManager':
##
##     version
```

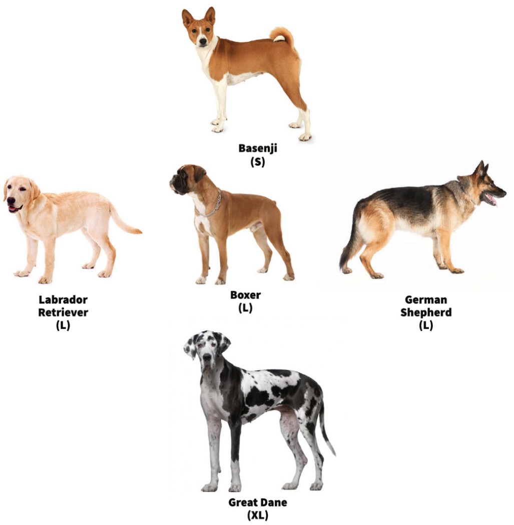
```r
#for msa pretty print
library("tinytex")
#visualization of results
library("ggplot2")
#for clustering of DNA seqs
library("DECIPHER")
```

```
## Loading required package: RSQLite

## Loading required package: parallel
```

```r
#for cleaning up dendograms
library('dendextend')
```

```
##
## ---------------------
## Welcome to dendextend version 1.15.2
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## ---------------------

##
## Attaching package: 'dendextend'

## The following object is masked from 'package:Biostrings':
##
##     nnodes

## The following object is masked from 'package:stats':
##
##     cutree
```

First we will be performing MSA and clustering with various sequence data (SNPS and gene fragments). Unfortunately, the scope of this data is limited to the 5 breeds shown below

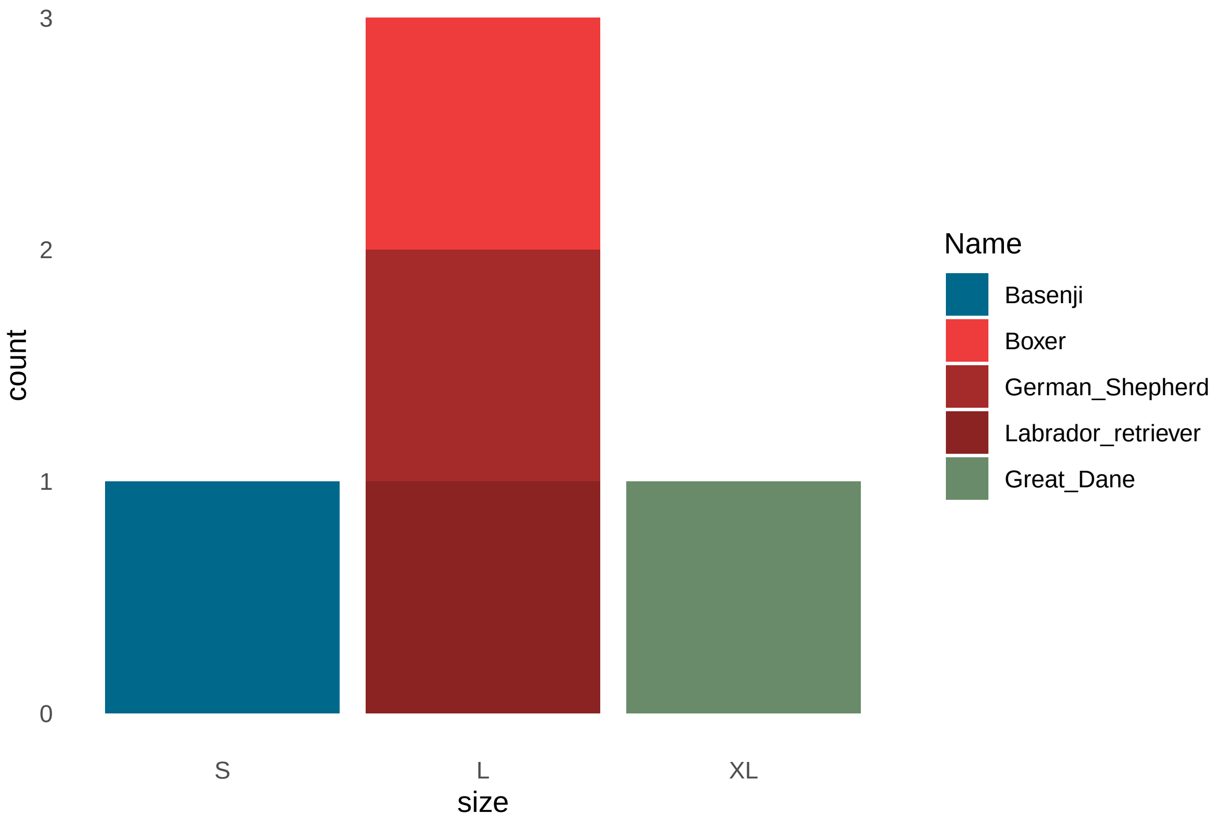


Function definitions

```
#global variable
alignment_name<<-""

#notebook functions

#align fasta from file_name with names from name file (visualization purposes) after alignment displays
mult_alingments<-function(file_name,fasta_names,name,big_aln=FALSE,dna_set=TRUE){

  #read in fasta for all dogs
  if(dna_set=="TRUE"){
  string_set<-readDNAStringSet(file=file_name,use.names=FALSE)
  }
  else{
      string_set<-readAAStringSet(file=file_name,use.names=FALSE)

  }

  #read in seq names as list
  table=read.table(fasta_names, header = FALSE, sep = "\n")[["V1"]]

  #update names for pretty print
  names(string_set)<-table

  #align unnamed seqs
  alignment<-msa(string_set,order="input")
  #if seq cant be display with msa pretty print, return
  if(big_aln==TRUE){
    return(alignment)
  }
  #update global variable so multiple pretty print runs dont overrun eachother
  alignment_name<<-gsub(" ", "", paste(name,".pdf"), fixed = TRUE)

  #return pretty alignment, does not show up on my console
  msaPrettyPrint(alignment, file=alignment_name,output="pdf",
                 showNames="right",showLogo="top",askForOverwrite=FALSE,
                 showNumbering="none",paperWidth=6,paperHeight=3)
return(alignment_name)
}
#have figure with white background, no gridline and only axis ticks, no lines
tune_figure<-function(fig){
  return(fig+theme_minimal()+theme(
    plot.background = element_blank(),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank()))
}
#create dendogram based on fasta files, names of items clustered in fasta_names
create_dendogram<-function(fasta_path, fasta_names, fig_title){
  #grab DNA info from coallated file
  dna <- string_set<-readDNAStringSet(file=fasta_path,use.names=FALSE)
  #get sequence names
  names(dna)=read.table(fasta_names, header = FALSE, sep = "\n")[["V1"]]
  #create distance matrix for clustering
  d1 <- DistanceMatrix(dna, type="dist")
  #form dendogram
  dendogram<-IdClusters(d1, method="complete", cutoff=0.05, showPlot=FALSE,type="dendrogram")
  #fix names being cut-off
  nodePar <- list(lab.cex = 0.6, pch = c(NA, 19),
                  cex = 0.7, col = "black")
#plot results
plot(as.dendrogram(dendogram), ylab = "Height", nodePar =nodePar,main=fig_title)
  return(as.dendrogram(dendogram))
}
```

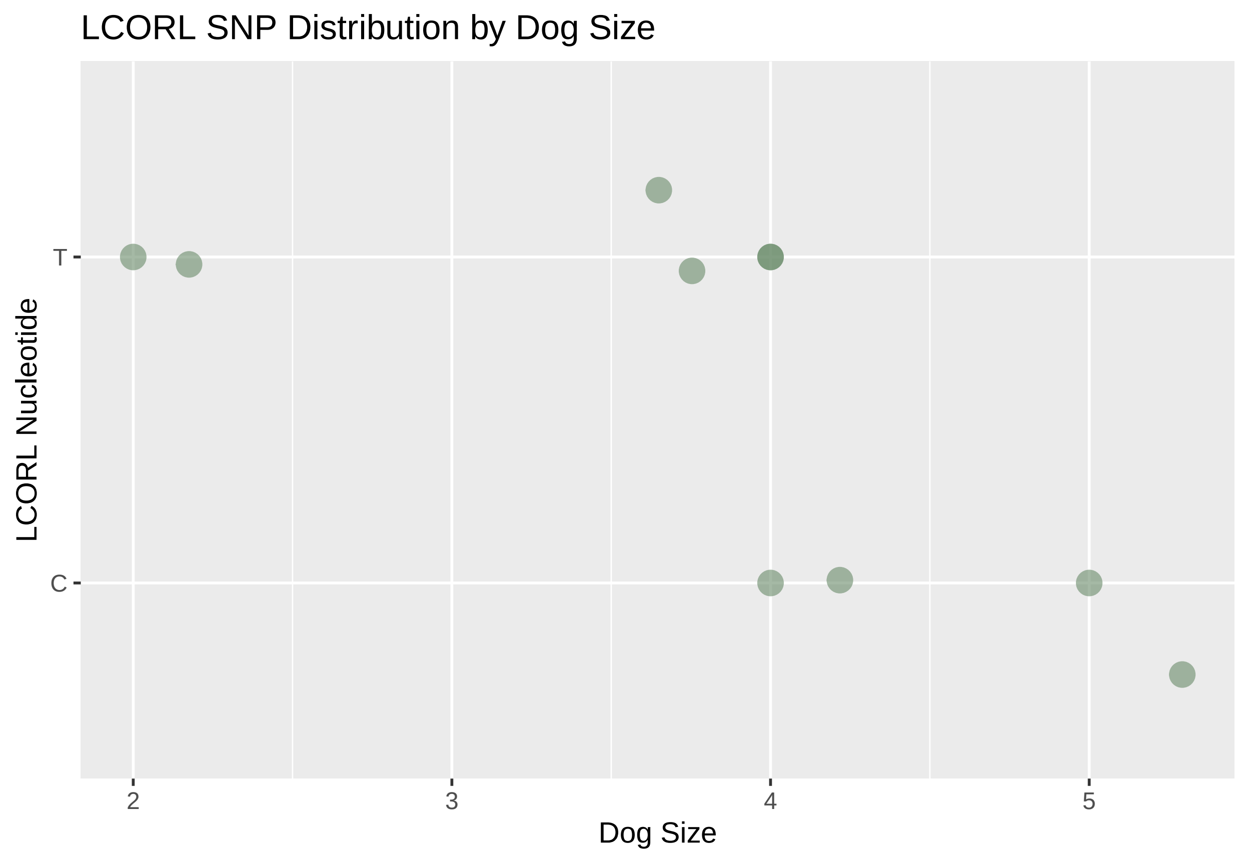
EDA of Sequence data

```
#visualize size breakdown of dogs
snps<-read_excel("dog snps.xlsx")
#fix ordering of legend
snps$Name <- factor(snps$Name, levels = c("Basenji", "Boxer", "German_Shepherd","Labrador_retriever","G
p<-ggplot(data = snps, aes(size))+scale_x_discrete(limits = c("S","L","XL"))+geom_bar(aes(fill = Name))+
tune_figure(p)
```



LCORL Sequence Analysis SNP scatterplot, see if any obvious trends

```
#visualize LCORL SNP by size
p<-ggplot(data = snps, mapping = aes(y=lcorl,x=size_num))+geom_point(size=4,alpha=0.6,color="darkseagre
p+geom_jitter(size=4,alpha=0.6,color="darkseagreen4")
```



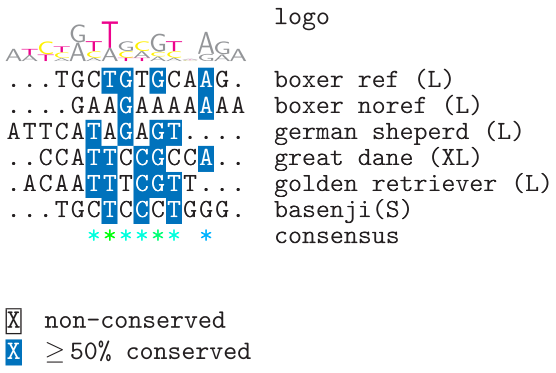
LCORL alignment, see if size-grouping is obvious

```
#LCORL CALL
alignment<-mult_alingments("fasta/LCORL_file.txt","fasta/names.txt","LCORL")
```

```
## use default substitution matrix
```

```
print(alignment_name)
```

```
## [1] "LCORL.pdf"
```



```
                 logo
...TGCTGTGCAAG.  boxer ref (L)
....GAAGAAAAAAA  boxer noref (L)
ATTCATAGAGT....  german sheperd (L)
..CCATTCCGCCA..  great dane (XL)
.ACAATTTCGTT...  golden retriever (L)
...TGCTCCCTGGG.  basenji(S)
     ****** *    consensus
```

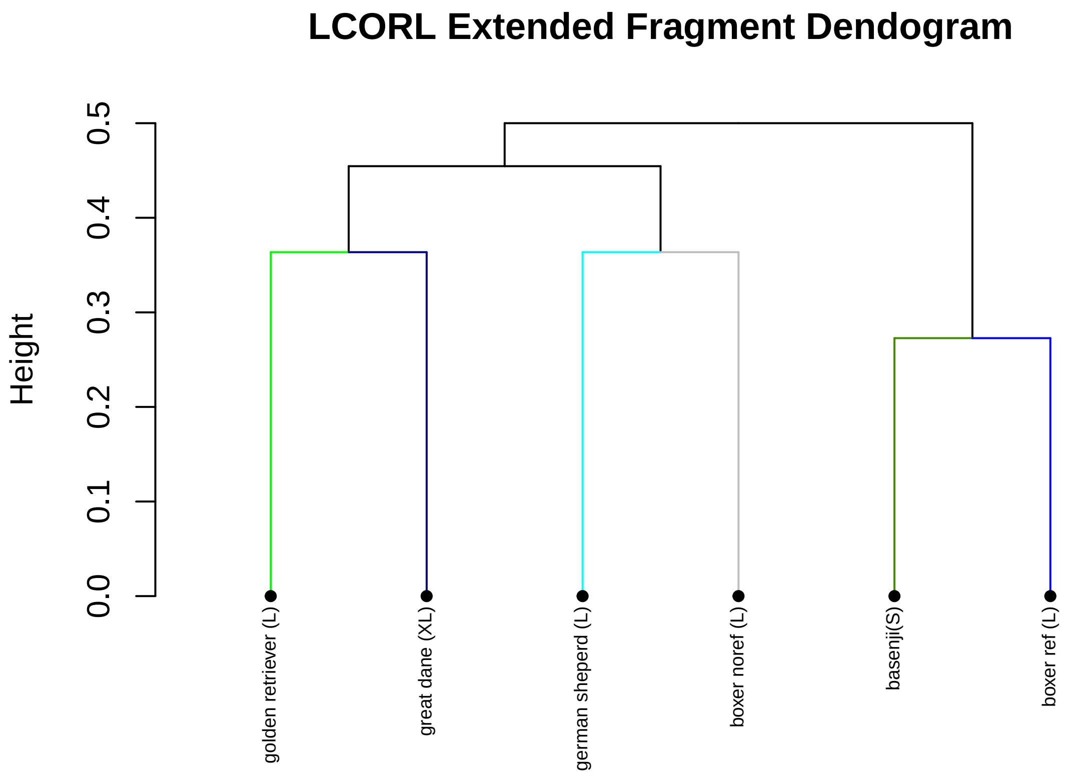
X non-conserved
X ≥50% conserved

LCORL

clusters, see if group by dog size

```
#Cluster LCORL extended fragment
create_dendogram("fasta/LCORL_file.txt", "fasta/names.txt", "LCORL Extended Fragment Dendogram")
```

```
## ================================================================================
##
## Time difference of 0 secs
##
## ================================================================================
##
## Time difference of 0.01 secs
```



```
## 'dendrogram' with 2 branches and 6 members total, at height 0.5
```

IGF1 ANALYSIS EDA of IGF1 SNP, see if are any obvious trendds

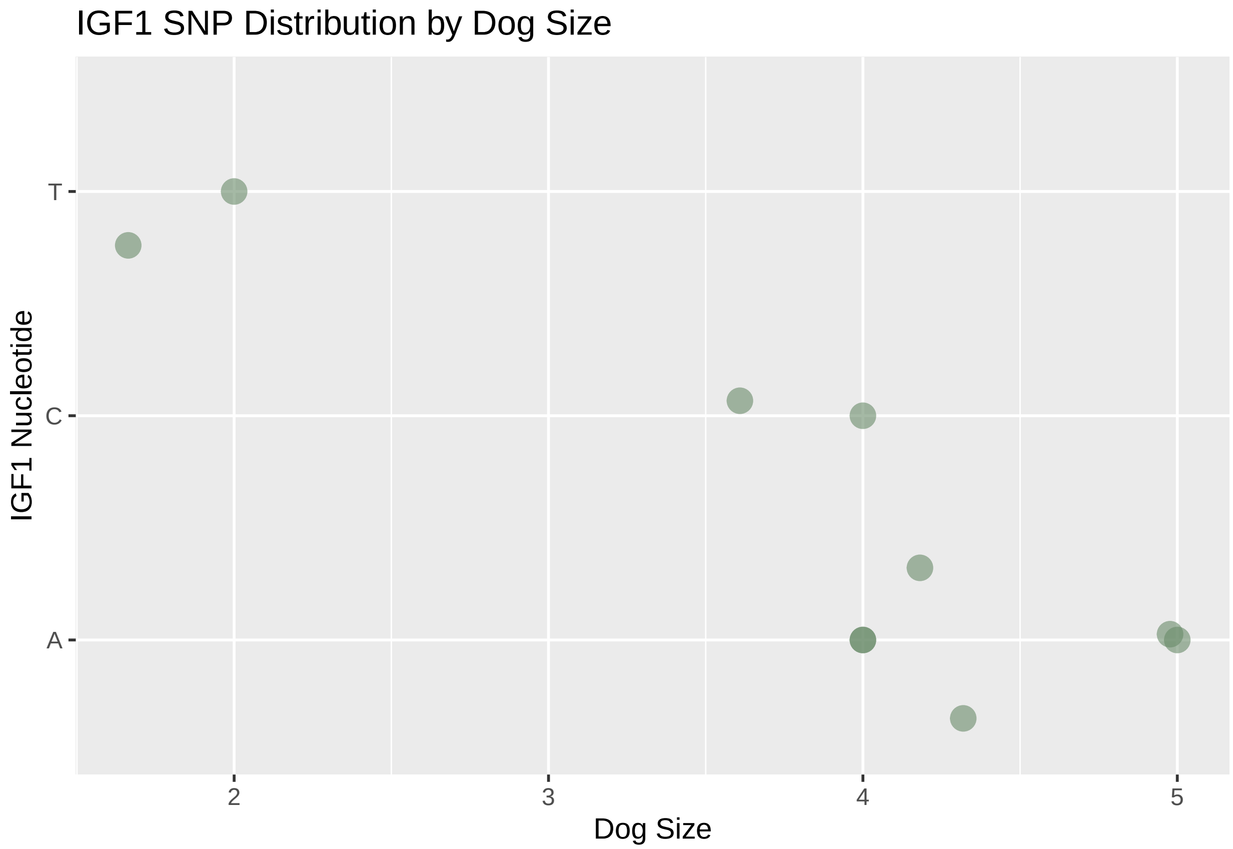
```
abbrev_x <- c("A","C","'G","'T")
print(length(abbrev_x))
```

```
## [1] 4
```

```
print(length(seq(0,4,by=1)))
```

```
## [1] 5
```

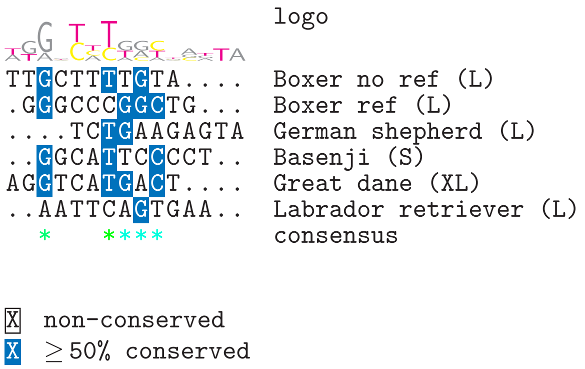
```
#visualize IGF1 SNP by size
p<-ggplot(data = snps, mapping = aes(y=igf1,x=size_num))+geom_point(size=4,alpha=0.6,color="darkseagreen
p+geom_jitter(size=4,alpha=0.6,color="darkseagreen4")
```



IGF1 Alignment, see if size-grouping is obvious

```r
#IGF1 CALL
alignment<-mult_alingments("fasta/igf1.fasta","fasta/igf1_names.txt","igf1")
```

```
## use default substitution matrix
```



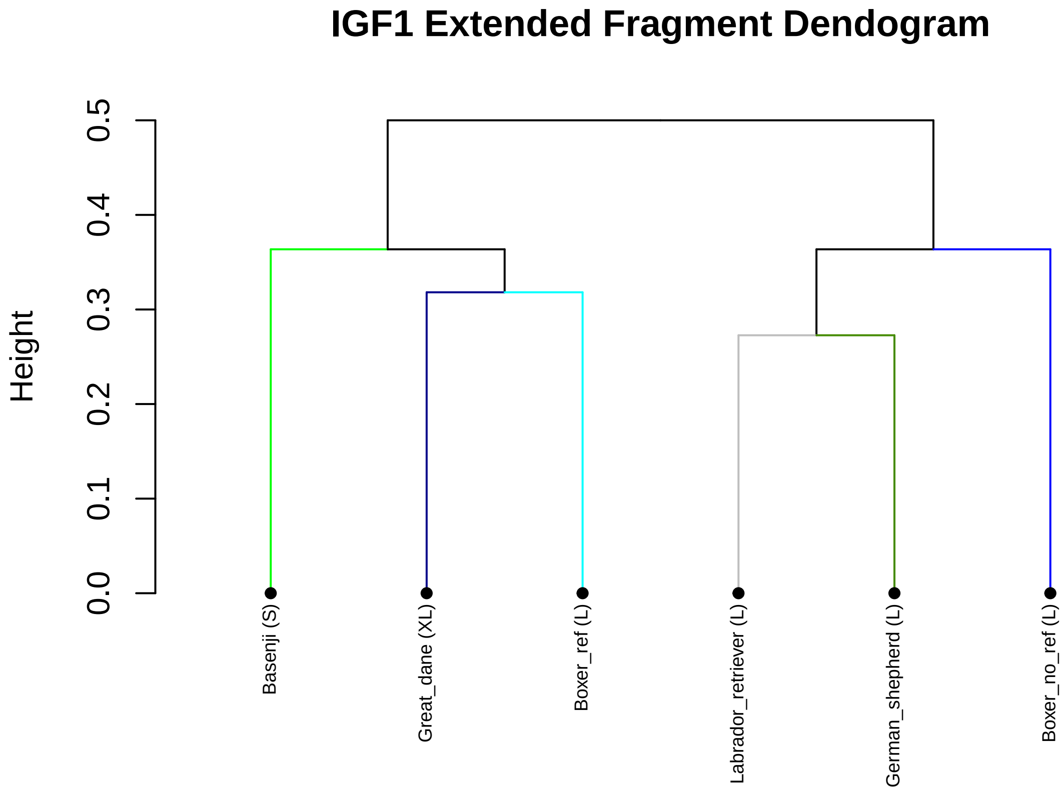
```
                  logo
TTGCTTTTGTA....   Boxer no ref (L)
.GGGCCCGGCTG...   Boxer ref (L)
....TCTGAAGAGTA   German shepherd (L)
..GGCATTCCCCT..   Basenji (S)
AGGTCATGACT....   Great dane (XL)
..AATTCAGTGAA..   Labrador retriever (L)
  *   ****        consensus


X  non-conserved
X  ≥50% conserved
```

http://www.sthda.com/english/wiki/beautiful-dendrogram-visualizations-in-r-5-must-known-methods-unsupervised-machine-learning#plot.dendrogram-function for look and non cut off stuff

IGF1 Clustering, see if size-grouping is obvious

```
#Cluster IGF1 extended fragment
create_dendogram("fasta/igf1.fasta", "fasta/igf1_names.txt", "IGF1 Extended Fragment Dendogram")
```

```
## ================================================================================
##
## Time difference of 0 secs
##
## ================================================================================
##
## Time difference of 0 secs
```



```
## 'dendrogram' with 2 branches and 6 members total, at height 0.5
```

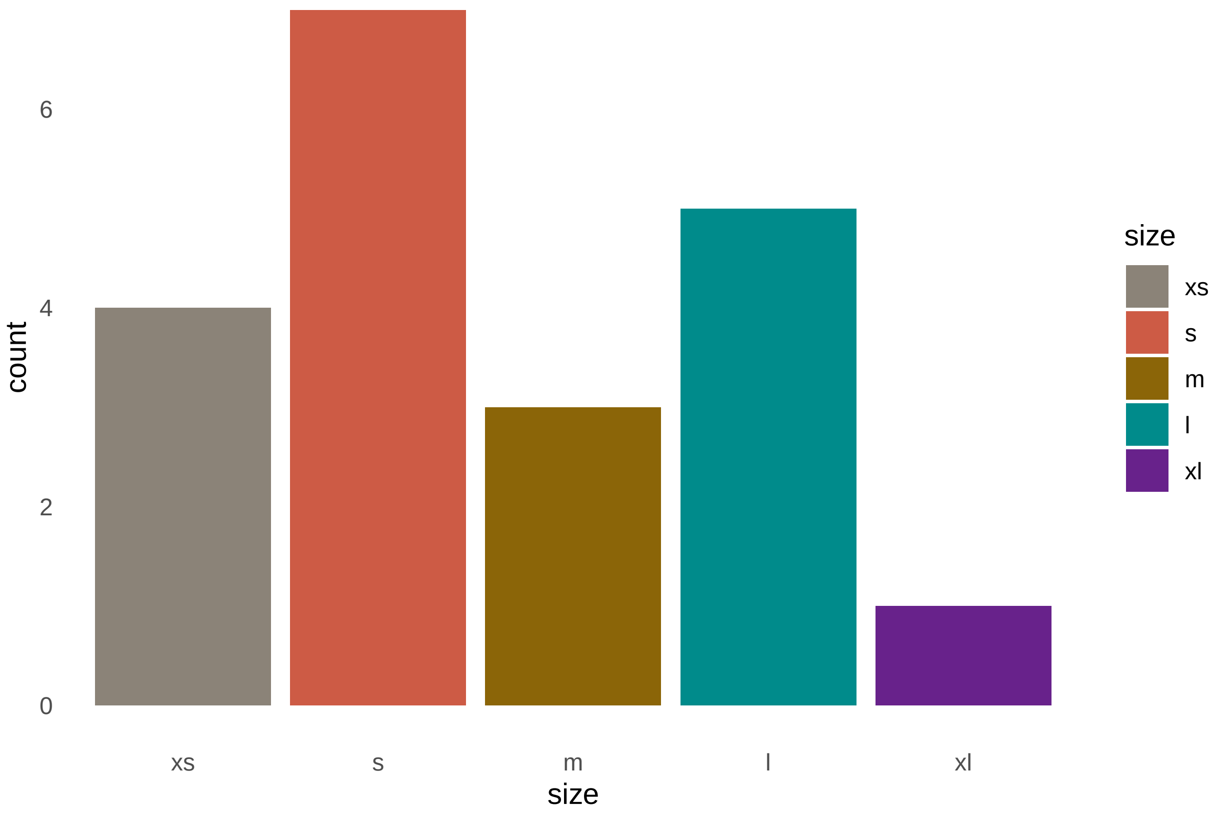
IGSF1 ANALYSIS, add back in for final submit

```
#library(seqinr)
#library(ape)

#takes too long to run, will add in final submission
#igsf1<-mult_alingments("fasta/IGSF1.fasta","fasta/IGSF1_names.txt","IGSF1",TRUE,FALSE)

#igsf1_aln <- msaConvert(igsf1, type="seqinr::alignment")
#d <- dist.alignment(igsf1_aln, "identity")
#dendogram<-IdClusters(d, method="complete", cutoff=0.05, showPlot=FALSE,type="dendrogram")
  #fix names being cut-off
 # nodePar <- list(lab.cex = 0.6, pch = c(NA, 19),
                #cex = 0.7, col = "black")
#plot results
#plot(as.dendrogram(dendogram), ylab = "Height", nodePar =nodePar,main="igsf1")
```

The differences sequence-wise seems to be minor when looking at such a narrow scope and small subsection of dogs. Thankfully there is also expression data for the genes studied which covers more dog-breeds, depicted below

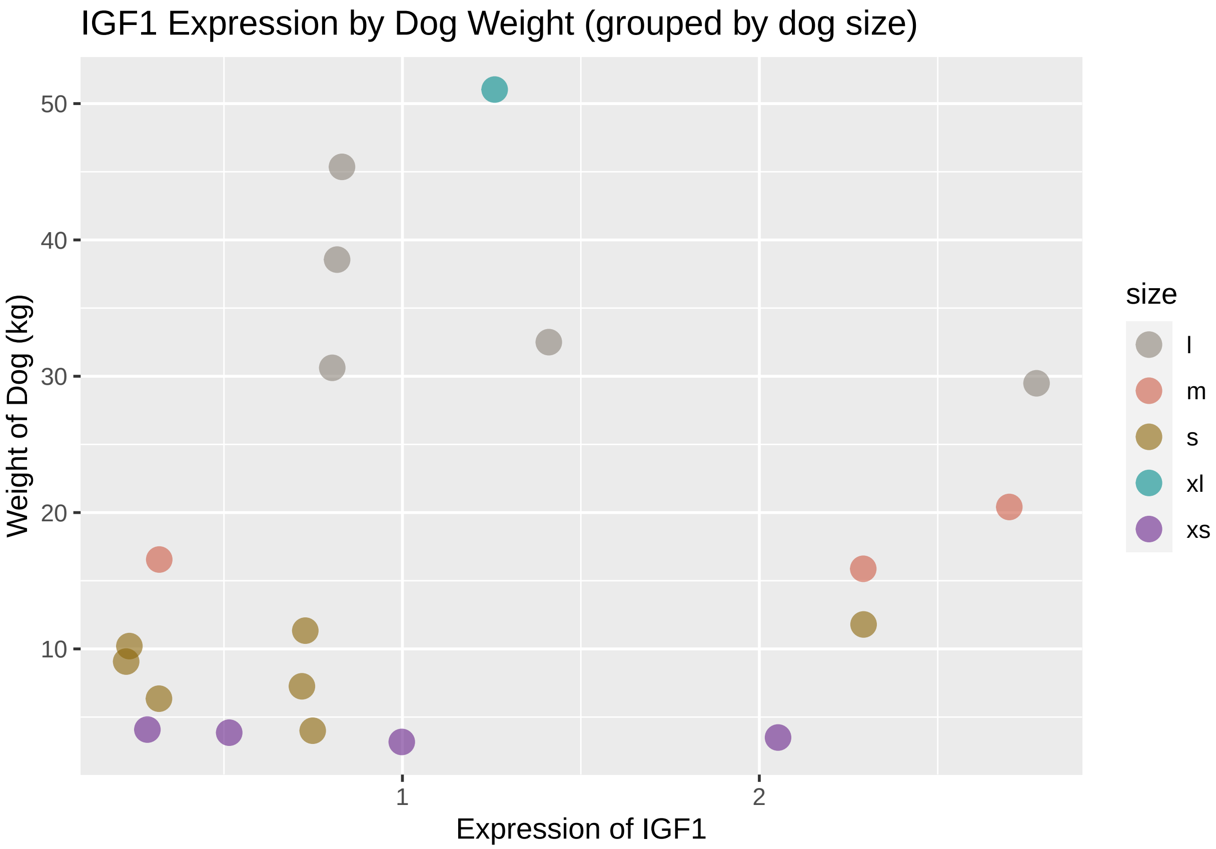
EDA of expression data, see size distribution of dogs

```
myColors <- c("antiquewhite4", " coral3", "darkgoldenrod4","darkcyan", "darkorchid4")

#visualize size breakdown of dogs
expression<-read_excel("dog snps.xlsx",sheet="IGF1")
#fix ordering of bars
expression$size <- factor(expression$size, levels = c("xs","s","m","l","xl"))

p<-ggplot(data = expression, aes(size))+geom_bar(aes(fill = size))+scale_fill_manual(values =myColors)
#clean up barplot
tune_figure(p)
```



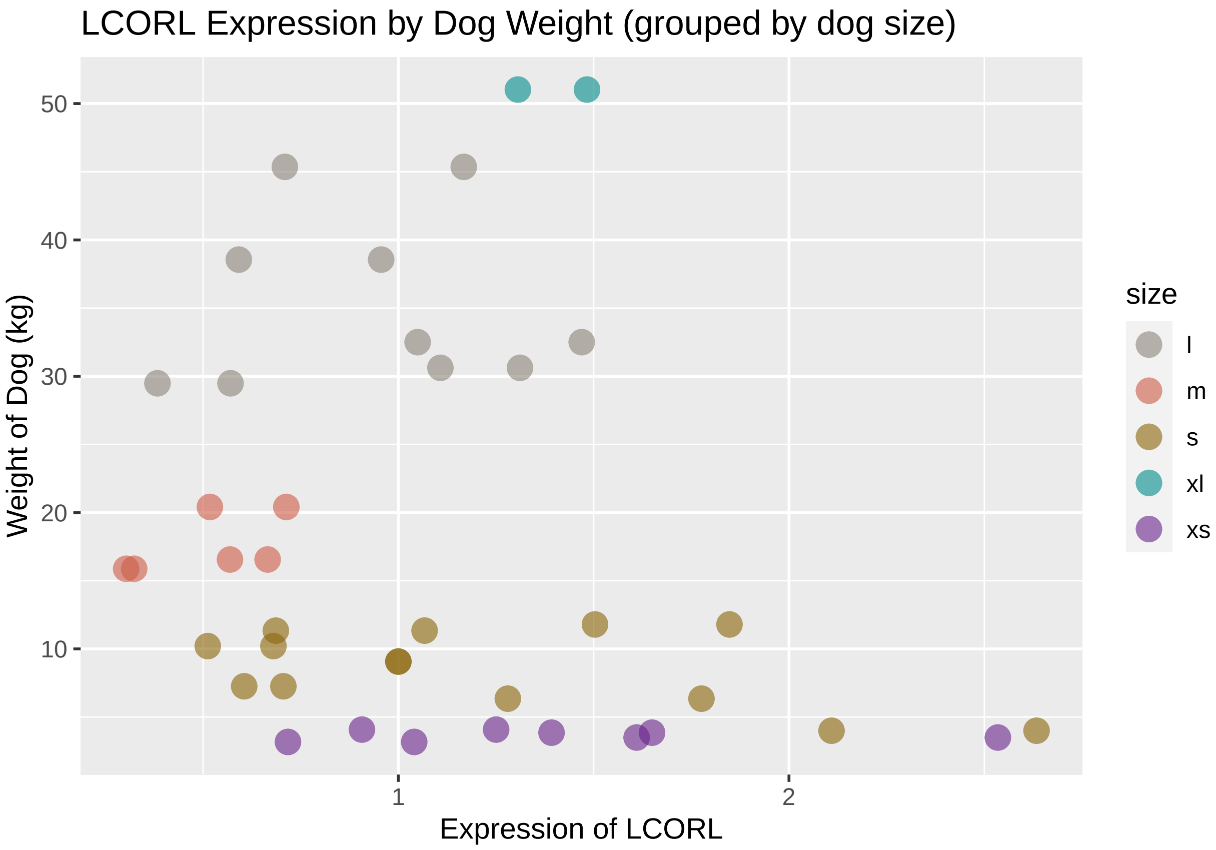
Distribution much more balanced than sequence data, but favors small dogs

```
#expression data
expression<-read_excel("dog snps.xlsx",sheet="IGF1")
myColors <- c("antiquewhite4", " coral3", "darkgoldenrod4","darkcyan", "darkorchid4")
p<-ggplot(data = expression, mapping = aes_string(x="norm_exp",y="weight_kg" ,col= "size"))+geom_point(
p+scale_color_manual(values=myColors)
```



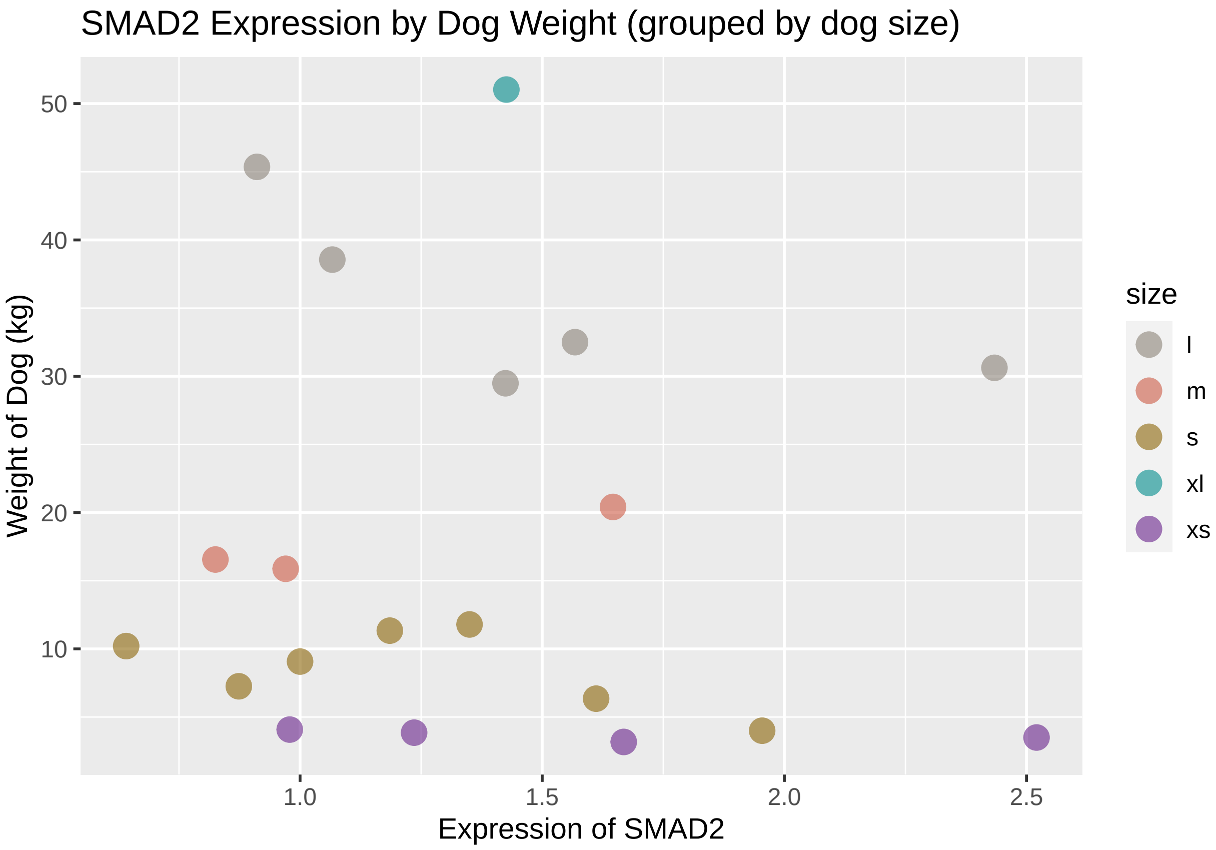
LCORL had two expression sets for each dog, so i included both

```
expression<-read_excel("dog snps.xlsx",sheet="LCORL")
p<-ggplot(data = expression, mapping = aes_string(x="norm_exp",y="weight_kg" ,col= "size"))+geom_point(
p+scale_color_manual(values=myColors)
```



igsf1 has no gene exp data so i swapped to smad2 which has similar function (smad2 is too long to align so it works out)

```
expression<-read_excel("dog snps.xlsx",sheet="SMAD2")
p<-ggplot(data = expression, mapping = aes_string(x="norm_exp",y="weight_kg" ,col= "size"))+geom_point(
p+scale_color_manual(values=myColors)
```


SMAD2 Expression by Dog Weight (grouped by dog size)
Weight of Dog (kg)
50
40
30
20
10
1.0
1.5
2.0
2.5
Expression of SMAD2
size
l
m
s
xl
xs