

st542_project10

Jingjing Li

2023-06-15

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.2      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2     3.4.2      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr       1.0.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
library(haven)
library(tidyverse)
library(ggplot2)
library(haven)
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.3.1
```

```
##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
##
##    %+%, alpha
```

Exploring the merged data

```
survey_merged <- read_dta("../data/merged_survey_panel196_19_noID.dta")
dim(survey_merged)
```

```
## [1] 55734 165
```

```

survey_merged_clean <- survey_merged %>%
  filter(!is.na(lotid_qualtrics))
# Removing duplicates for modelling - forgot to ask Mariana which should be kept, so just keeping first
# Also open to just dropping all duplicated lotids - there's only 42
survey_final <- survey_merged_clean %>% distinct(lotid_qualtrics, .keep_all = TRUE)

```

Now to check for nulls in columns:

```

missing_vals <- survey_final %>% summarise_all(~sum(is.na(.)))
t(missing_vals)

```

```

##                [,1]
## technician_times 2284
## family           1026
## north            1026
## northeast        1026
## southeast        1026
## south            1026
## lotid_qualtrics    0
## studycode        1026
## havecattle        1131
## havecattleother   1147
## cattleherd        1184
## cattlemilk        1152
## milkhardry        1651
## milkharwet        1651
## milkhardry_min_dairy 1167
## milkharwet_max_dairy 1166
## pmilkdrynominal   1825
## pmilkwetnominal   1771
## milkinwet         1226
## milkindry         1226
## irrigation_pas    2251
## drought_year      2032
## risk              1086
## annual_crops      1026
## perennial_crops   1026
## incbeef           1577
## cattle_price       2150
## incpension         1058
## incbolsafam        1058
## incoff             1182
## lotprice           1098
## housecity          1026
## keep_veg           1026
## soiltype_sand      1194
## soiltype_silt      1194
## soiltype_clay      1194
## soiltype_other     1194
## soiltype_dontknow  1194
## fishtanks_year    2062
## reservoir_year     2256
## trough_year        2099

```

## dam_year	1751
## milk_room_year	2236
## milktanks_lot_year	2180
## irrigation_year	2255
## well_year	1464
## caixa_seca_year	2227
## fishtanks_have	1372
## reservoir_have	1436
## trough_have	1392
## dam_have	1281
## milk_room_have	1438
## well_have	1148
## caixa_seca_have	1467
## vechval	1026
## loan	1853
## loan_investment	1853
## unions	1026
## documents	1026
## aveeduhh	1098
## fert_pasture	1026
## pest_pasture	1026
## cattleinputs_breed	1026
## semiconfine	1250
## fallow	1132
## soil_analysis	1096
## pasture	1026
## annuals	1026
## perennials	1026
## forest	1026
## yearmove	1034
## techvisit	1039
## sellcattle_droughtexp	1971
## waterstructure	1026
## feed_cattle	1026
## pasture_productivity	1026
## forest_rec	1026
## fish_bee	1026
## milktank_have	1026
## yearmig	2347
## inchf	2347
## inccalf	2347
## incpigs	2347
## incchicken	2347
## incsheepgoat	2347
## inchorses	2347
## incmules	2347
## incdonkey	2347
## incotherlive	2347
## valcattle	2347
## plows	2347
## central	2347
## pricebeef	2347
## incbolsaescola	2347
## mow	2347

## landsold_year	2347
## landbuy_year	2347
## honey_har	2347
## honey_price	2347
## fish_har	2347
## workers	2347
## PIgrass_cattle	0
## PIbuilding_cattle	0
## PIsilage_cattle	0
## PImowing_cattle	0
## soil	2347
## aveslope	2347
## distopo	2347
## opo_ttmin	2347
## jiparana_ttmin	2347
## closest_ttmin	2347
## pmilkdryreal	2347
## pmilkwetreal	2347
## rainfallmin_year	1026
## rainfallmax_year	1026
## rainfall_wet6	1026
## rainfall_dry6	1026
## rainfall_dry	1026
## rainfall_wet	1026
## rainfall_year	1026
## rainfall	1026
## rainfall_jan	1026
## rainfall_feb	1026
## rainfall_mar	1026
## rainfall_apr	1026
## rainfall_may	1026
## rainfall_june	1026
## rainfall_july	1026
## rainfall_aug	1026
## rainfall_sep	1026
## rainfall_oct	1026
## rainfall_nov	1026
## rainfall_dec	1026
## SPImin_year	1026
## SPImax_year	1026
## SPI_wet6	1026
## SPI_dry6	1026
## SPI_dry	1026
## SPI_wet	1026
## SPI_year	1026
## SPI_jan	1026
## SPI_feb	1026
## SPI_mar	1026
## SPI_apr	1026
## SPI_may	1026
## SPI_june	1027
## SPI_july	1113
## SPI_aug	1026
## SPI_sep	1026

```
## SPI_oct          1026
## SPI_nov          1026
## SPI_dec          1026
## off_farm         1182
## unions_part      1026
## cleared_area     1029
## cleared_area_fraction 1029
## Drainage_AreaKM_Ari 1970
## ARIQ_drainage     1017
## OPO_area_km       1869
## OPO_drainage       982
## RO_area_km        1858
## C                 2347
## Rolim_drainage     886
## lotsize_GIS_ponds  0
## ponds_2019        0
```

There seem to be a lot of columns missing 1026 values, which puts the numbers back to what we saw in the original data. Clearly there is a lot of missing data across all areas, so I don't think we should drop all rows missing data.

Feature Engineering

Creating One Drainage Area Column

I will create one drainage area column, and then will drop all rows that don't have drainage area. Not sure if this is the best approach but this data is so messy!

```
# There are two lots that have areas in each region - will drop them
duplicate_regions <- survey_final %>% filter(!is.na(OPO_area_km) & !is.na(RO_area_km)) %>% select(lotid,
survey_final <- survey_final %>% filter(!(lotid_qualtrics %in% duplicate_regions$lotid_qualtrics))
survey_final <- survey_final %>% mutate(drainage_area_km = ifelse(!is.na(Drainage_AreaKM_Ari), Drainage_AreaKM_Ari,
region = ifelse(studycode == 1, "Ariquemes",
ifelse(studycode == 2, "Ouro Preto do Oeste", "Other")),
select(-c(Drainage_AreaKM_Ari, RO_area_km, OPO_area_km))
```

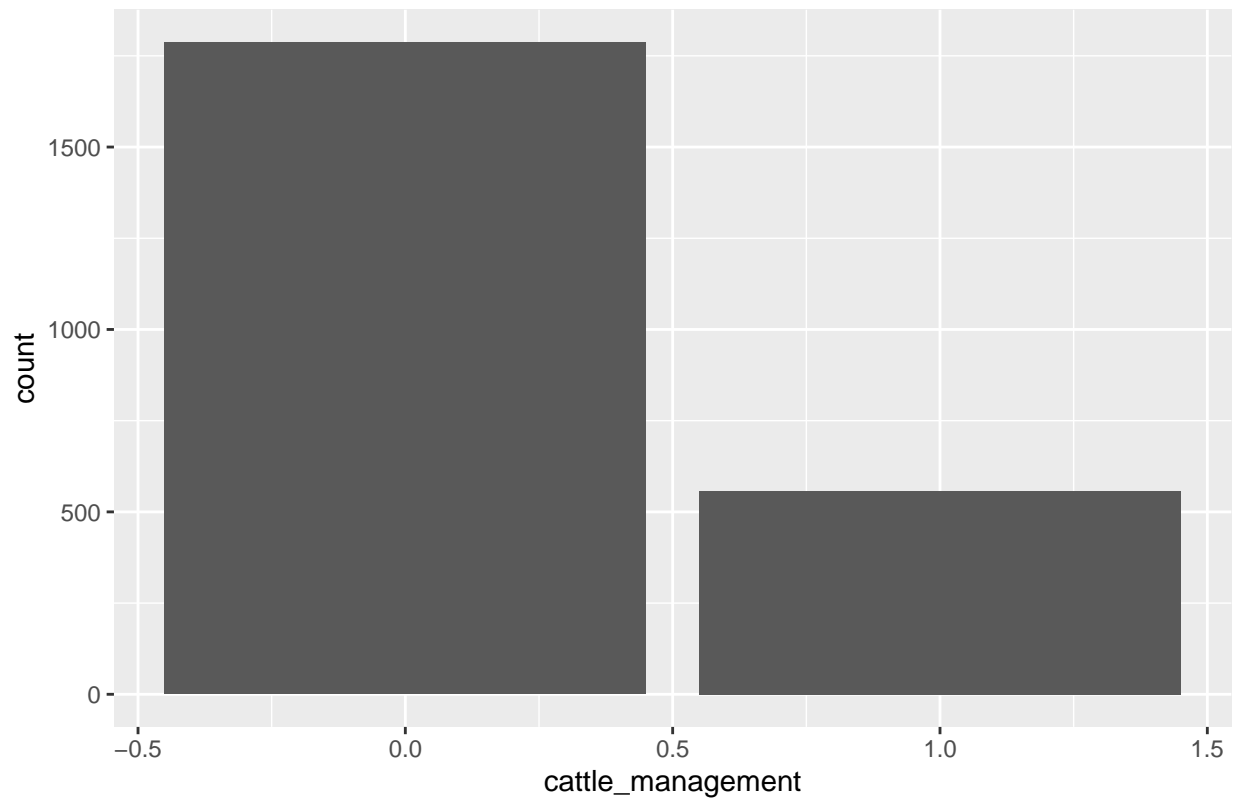
Creating Individual Adaptation Method Booleans

First, I need to create the individual adaptation method booleans i.e. `cattle_management` if the farmer employed any cattle management strategy. Then I can combine these to make a general adaptation method boolean.

```
survey_final <- survey_final %>% mutate(cattle_management = case_when(feed_cattle == 1 | soil_analysis == 1 |
pasture_management = case_when(irrigation_pas == 1 | pasture_management == 1 |
forest_conservation = case_when(forest_rec == 1 | keep_veg == 1 |
water_management = case_when(waterstructure == 1 | well_have == 1 |

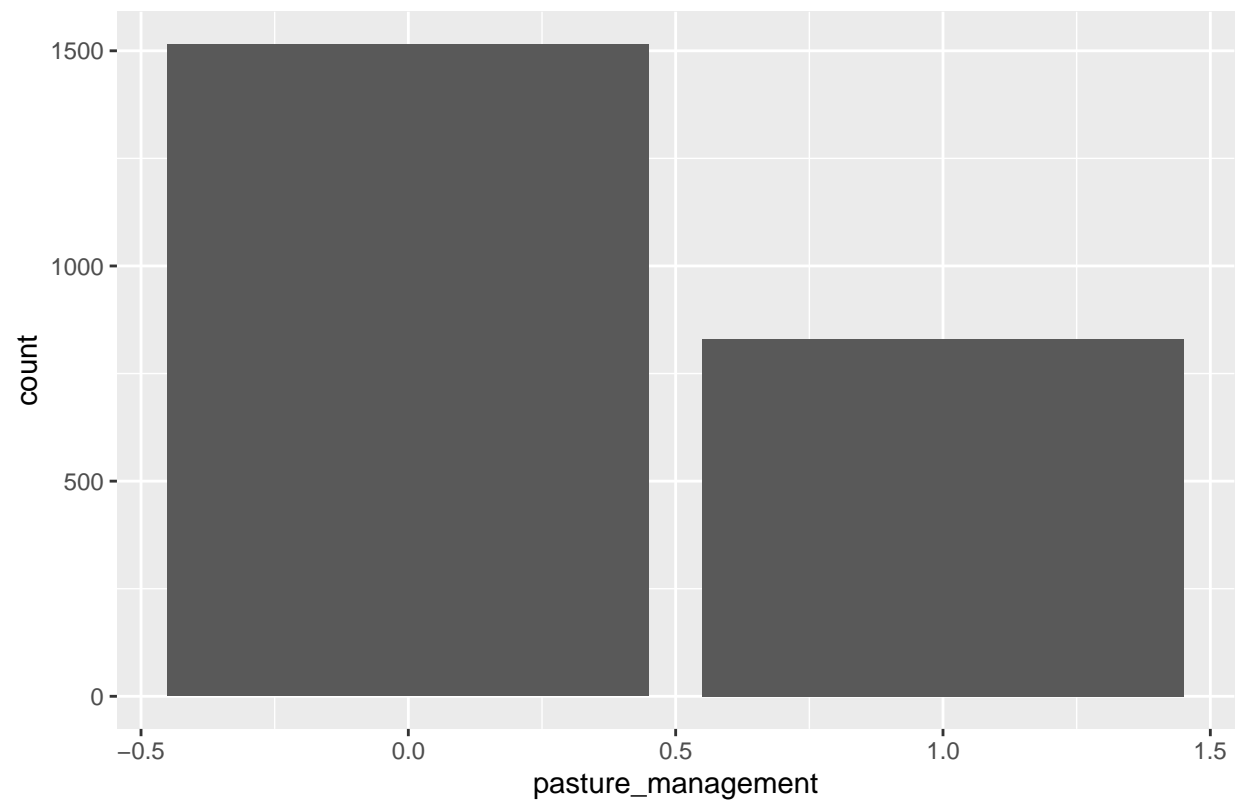
# Looking at distributions
g <- ggplot(data = survey_final, aes(x = cattle_management))
g + geom_bar() + labs(title = "Bar Plot of Cattle Management")
```

Bar Plot of Cattle Management

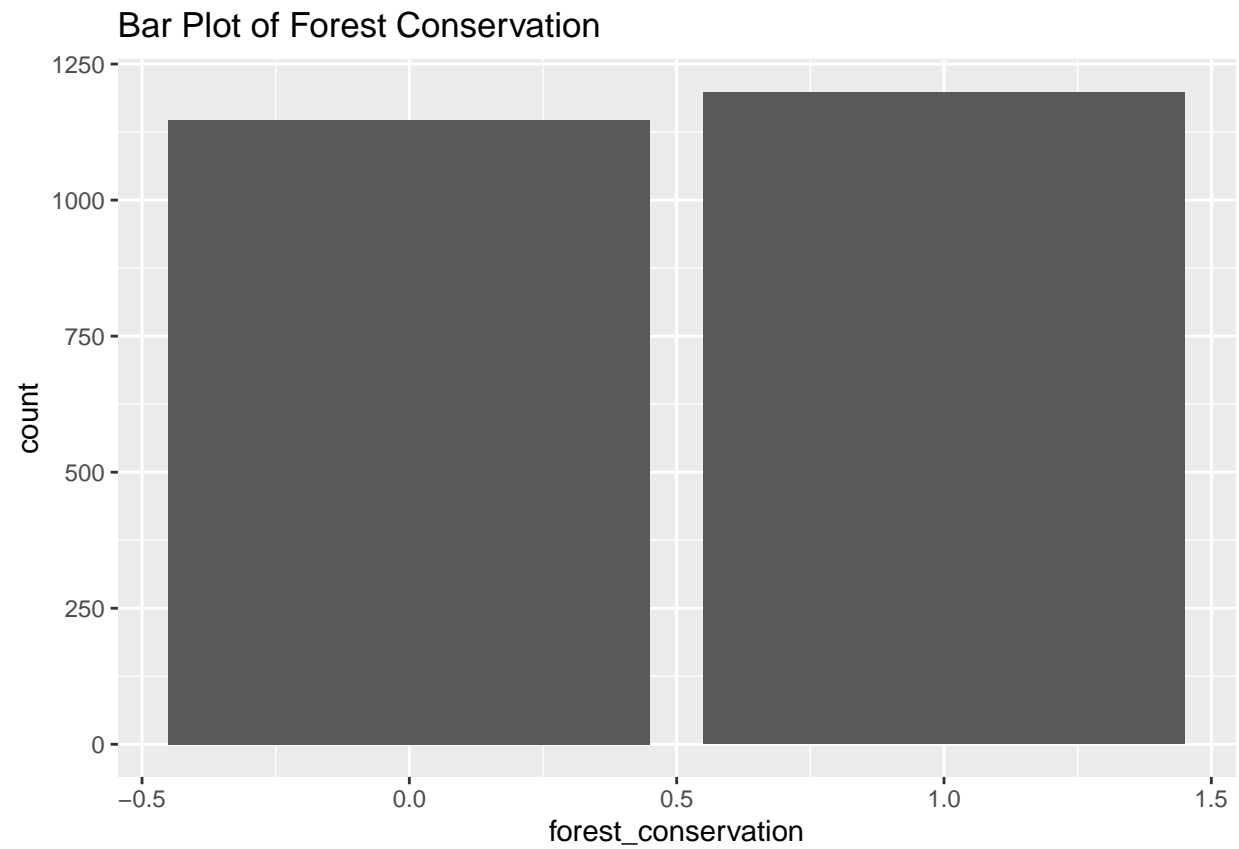


```
g <- ggplot(data = survey_final, aes(x = pasture_management))  
g + geom_bar() + labs(title = "Bar Plot of Pasture Management")
```

Bar Plot of Pasture Management

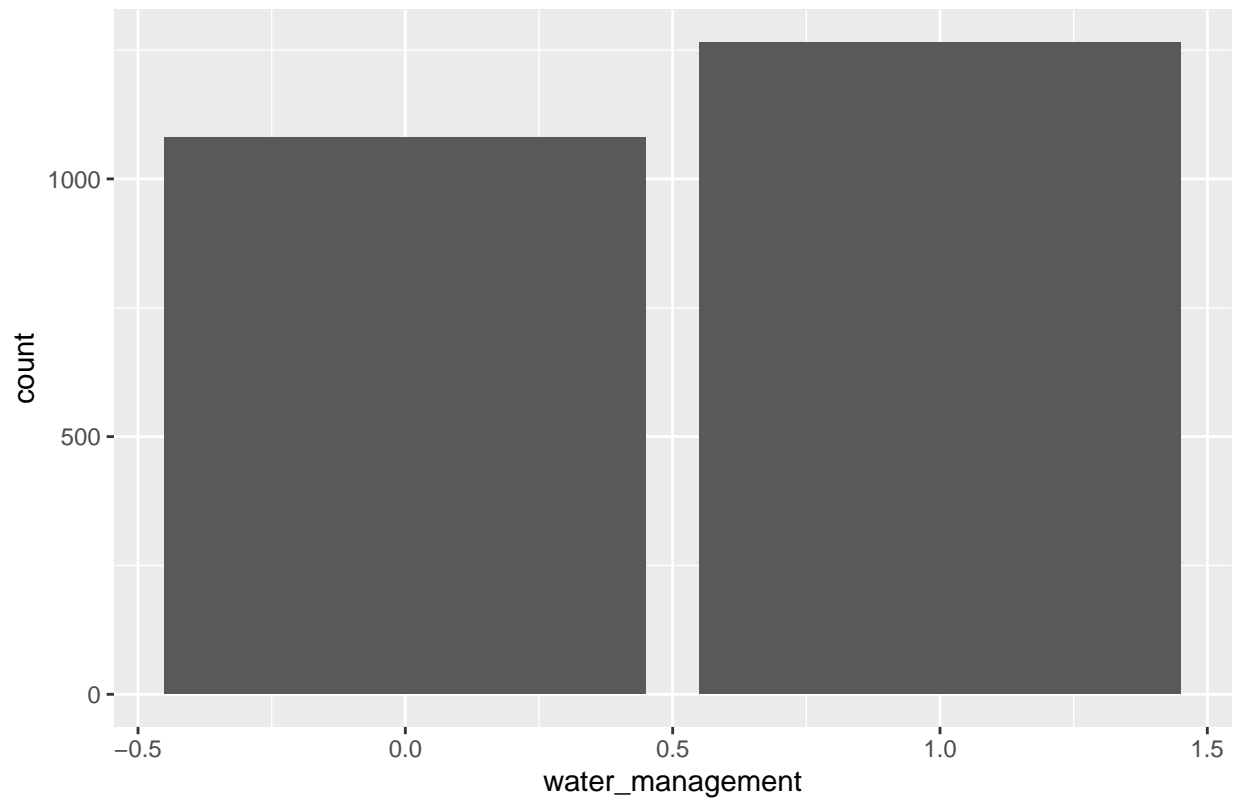


```
g <- ggplot(data = survey_final, aes(x = forest_conservation))  
g + geom_bar() + labs(title = "Bar Plot of Forest Conservation")
```



```
g <- ggplot(data = survey_final, aes(x = water_management))  
g + geom_bar() + labs(title = "Bar Plot of Water Management")
```


Bar Plot of Water Management



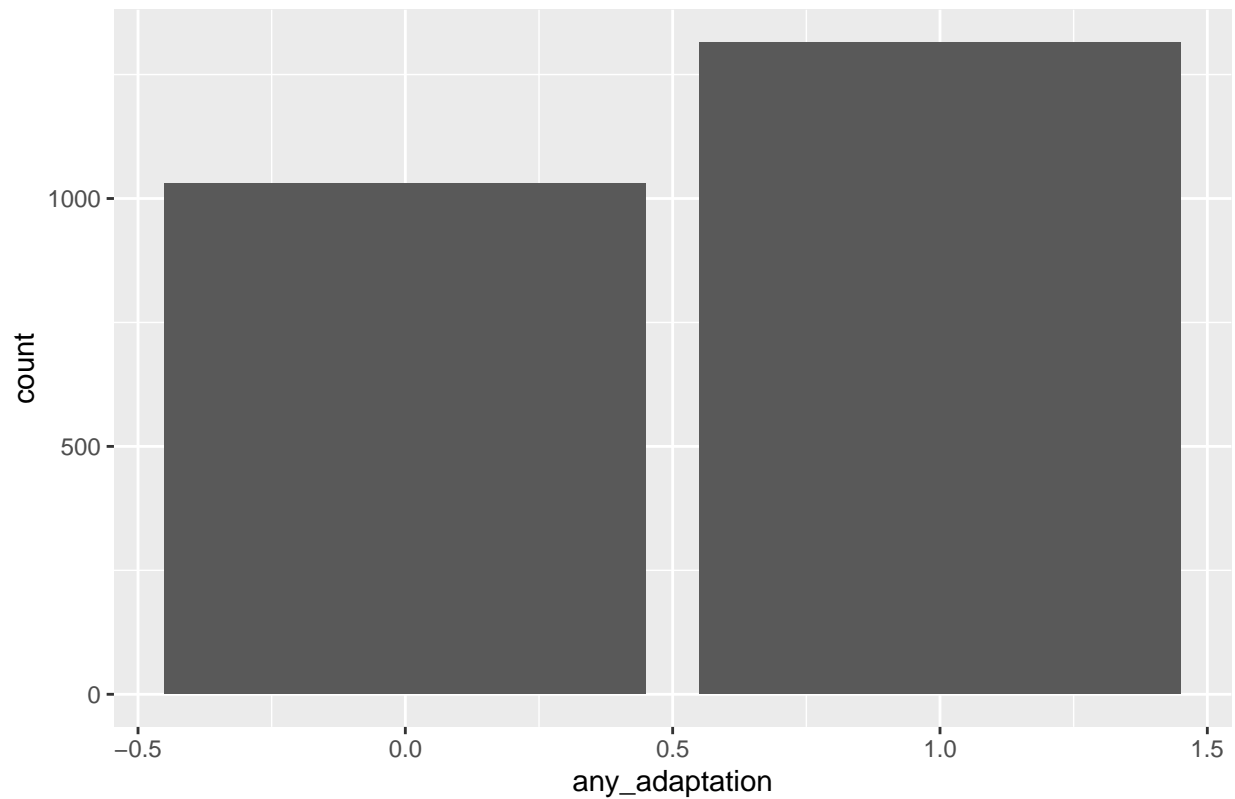
Creating General Adaptation Method Boolean

Now that the individual adaptation measures have been created, I will create a general adaptation method boolean

```
survey_final <- survey_final %>% mutate(any_adaptation = ifelse(cattle_management == 1 | pasture_management == 1, TRUE, FALSE))

# Looking at the distribution
g <- ggplot(data = survey_final, aes(x = any_adaptation))
g + geom_bar() + labs(title = "Bar Plot of Any Adaptation")
```

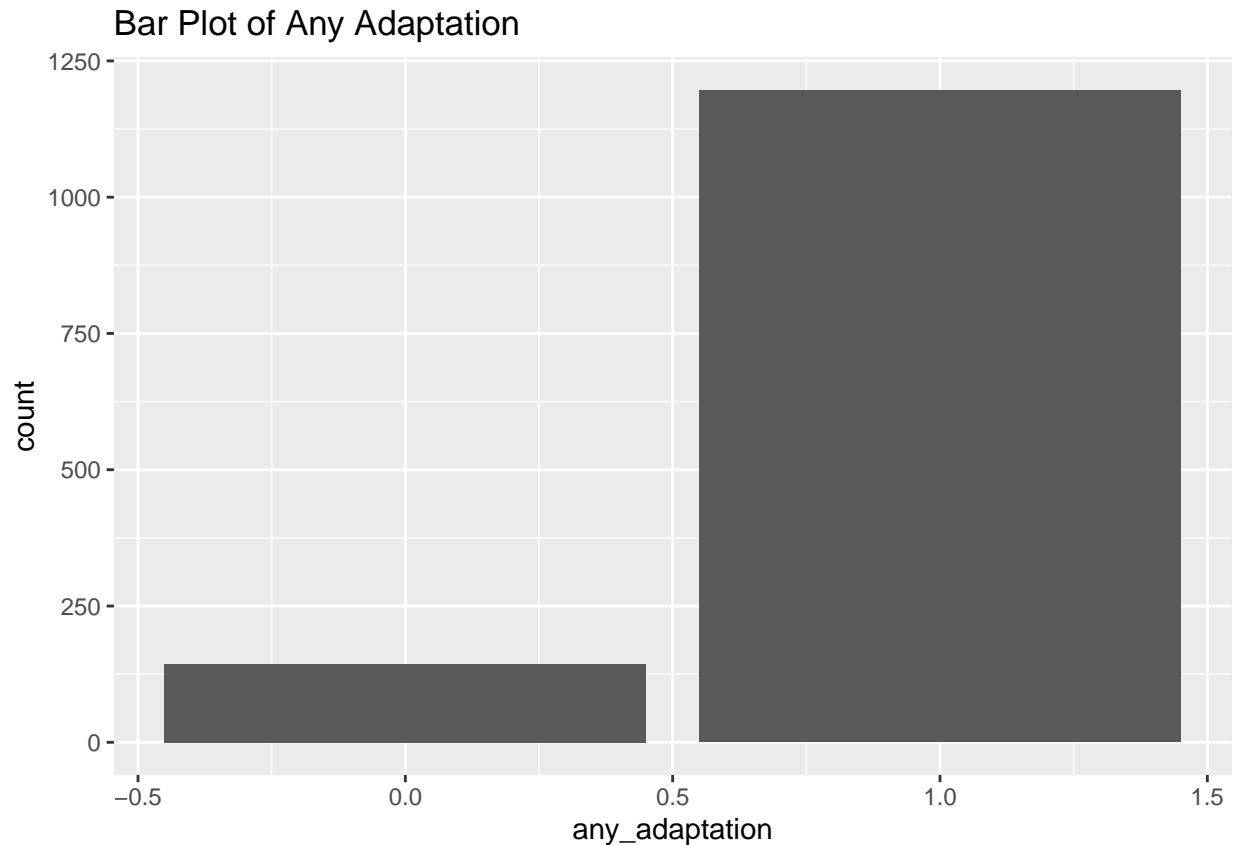
Bar Plot of Any Adaptation



Not a fully even data set, but better balanced than I predicted it would be. I don't think rebalancing is needed for the final analysis.

Now let's see if we remove farmers who don't have GIS information

```
survey_final_gis <- survey_final %>% filter(!is.na(drainage_area_km))  
  
# Looking at the distribution  
g <- ggplot(data = survey_final_gis, aes(x = any_adaptation))  
g + geom_bar() + labs(title = "Bar Plot of Any Adaptation")
```



Now it is very imbalanced. Not as horrible as it could be - but could affect the outputs of the model. Is SPI still missing for these?

```
missing_vals_gis <- survey_final_gis %>% summarise_all(~sum(is.na(.)))
t(missing_vals_gis)
```

```
##           [,1]
## technician_times 1280
## family           140
## north            140
## northeast        140
## southeast        140
## south            140
## lotid_qualtrics   0
## studycode        140
## havecattle        234
## havecattleother   247
## cattleherd        276
## cattlemilk        249
## milkhardry        725
## milkharwet        725
## milkhardry_min_dairy 262
## milkharwet_max_dairy 262
## pmilkdrynominal   858
## pmilkwetnominal   807
## milkinbwet        320
```

## milkindry	320
## irrigation_pas	1262
## drought_year	1051
## risk	194
## annual_crops	140
## perennial_crops	140
## incbeef	642
## cattle_price	1162
## incpension	168
## incbolsafam	168
## incoff	284
## lotprice	203
## housecity	140
## keep_veg	140
## soiltype_sand	293
## soiltype_silt	293
## soiltype_clay	293
## soiltype_other	293
## soiltype_dontknow	293
## fishtanks_year	1093
## reservoir_year	1259
## trough_year	1114
## dam_year	802
## milk_room_year	1237
## milktanks_lot_year	1188
## irrigation_year	1264
## well_year	536
## caixa_seca_year	1229
## fishtanks_have	469
## reservoir_have	527
## trough_have	485
## dam_have	377
## milk_room_have	528
## well_have	256
## caixa_seca_have	559
## vechval	140
## loan	889
## loan_investment	889
## unions	140
## documents	140
## aveeduhh	208
## fert_pasture	140
## pest_pasture	140
## cattleinputs_breed	140
## semiconfine	345
## fallow	235
## soil_analysis	202
## pasture	140
## annuals	140
## perennials	140
## forest	140
## yearmove	146
## techvisit	153
## sellcattle_droughtexp	996

## waterstructure	140
## feed_cattle	140
## pasture_productivity	140
## forest_rec	140
## fish_bee	140
## milktank_have	140
## yearmig	1340
## inchf	1340
## inccalf	1340
## incpigs	1340
## incchicken	1340
## incsheepgoat	1340
## inchorses	1340
## incmules	1340
## incdonkey	1340
## incotherlive	1340
## valcattle	1340
## plows	1340
## central	1340
## pricebeef	1340
## incbolsaescola	1340
## mow	1340
## landsold_year	1340
## landbuy_year	1340
## honey_har	1340
## honey_price	1340
## fish_har	1340
## workers	1340
## PIgrass_cattle	0
## PIbuilding_cattle	0
## PIsilage_cattle	0
## PImowing_cattle	0
## soil	1340
## aveslope	1340
## distopo	1340
## opo_ttmin	1340
## jiparana_ttmin	1340
## closest_ttmin	1340
## pmilkdryreal	1340
## pmilkwetreal	1340
## rainfallmin_year	140
## rainfallmax_year	140
## rainfall_wet6	140
## rainfall_dry6	140
## rainfall_dry	140
## rainfall_wet	140
## rainfall_year	140
## rainfall	140
## rainfall_jan	140
## rainfall_feb	140
## rainfall_mar	140
## rainfall_apr	140
## rainfall_may	140
## rainfall_june	140

```

## rainfall_july      140
## rainfall_aug       140
## rainfall_sep       140
## rainfall_oct       140
## rainfall_nov       140
## rainfall_dec       140
## SPImin_year        140
## SPImax_year        140
## SPI_wet6           140
## SPI_dry6           140
## SPI_dry            140
## SPI_wet            140
## SPI_year           140
## SPI_jan            140
## SPI_feb            140
## SPI_mar            140
## SPI_apr            140
## SPI_may            140
## SPI_june           141
## SPI_july           209
## SPI_aug            140
## SPI_sep            140
## SPI_oct            140
## SPI_nov            140
## SPI_dec            140
## off_farm           284
## unions_part        140
## cleared_area       143
## cleared_area_fraction 143
## ARIQ_drainage       131
## OPO_drainage        96
## C                  1340
## Rolim_drainage      0
## lotsize_GIS_ponds   0
## ponds_2019          0
## drainage_area_km    0
## region              140
## cattle_management   0
## pasture_management  0
## forest_conservation 0
## water_management    0
## any_adaptation      0

```

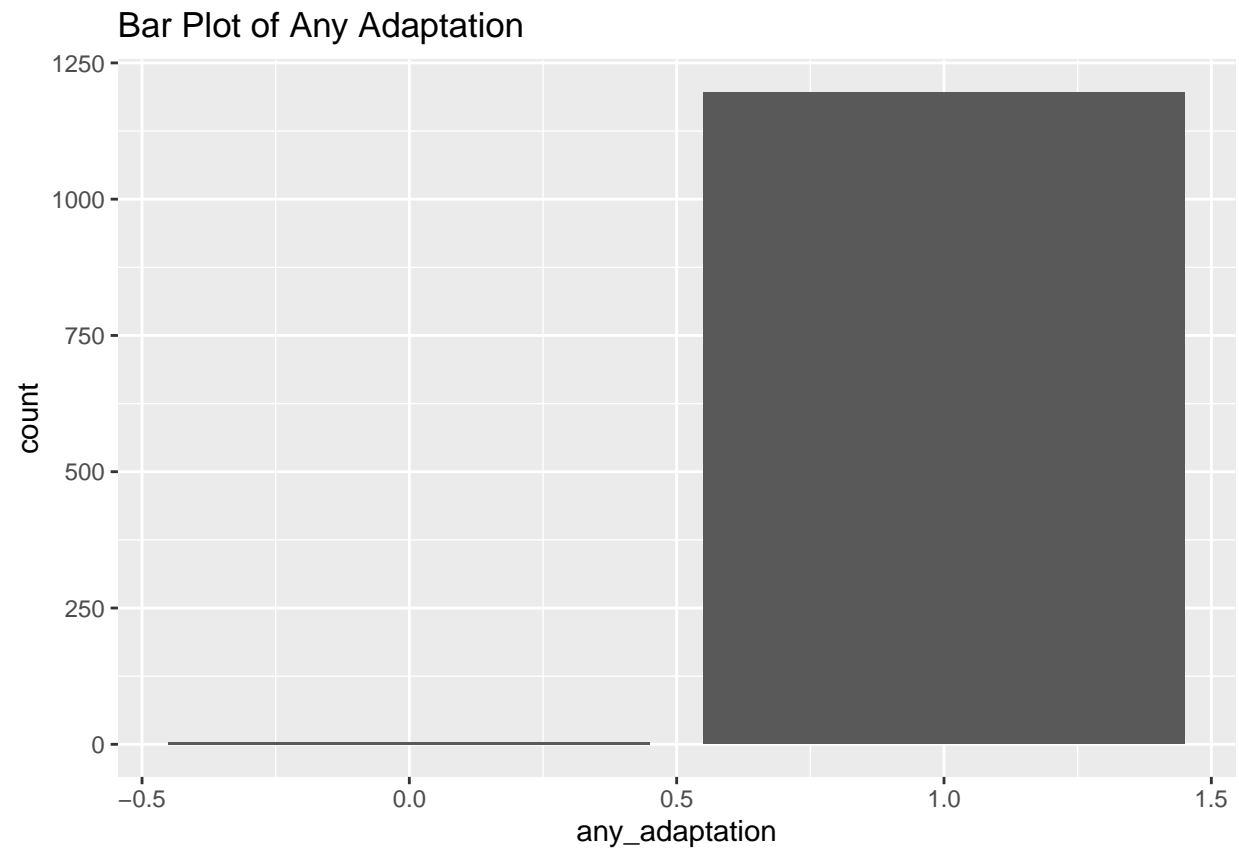
Now only for 140 of them, who are also missing region. Let's drop those and see:

```

survey_final_gis_clean <- survey_final_gis %>% filter(!is.na(SPImin_year))

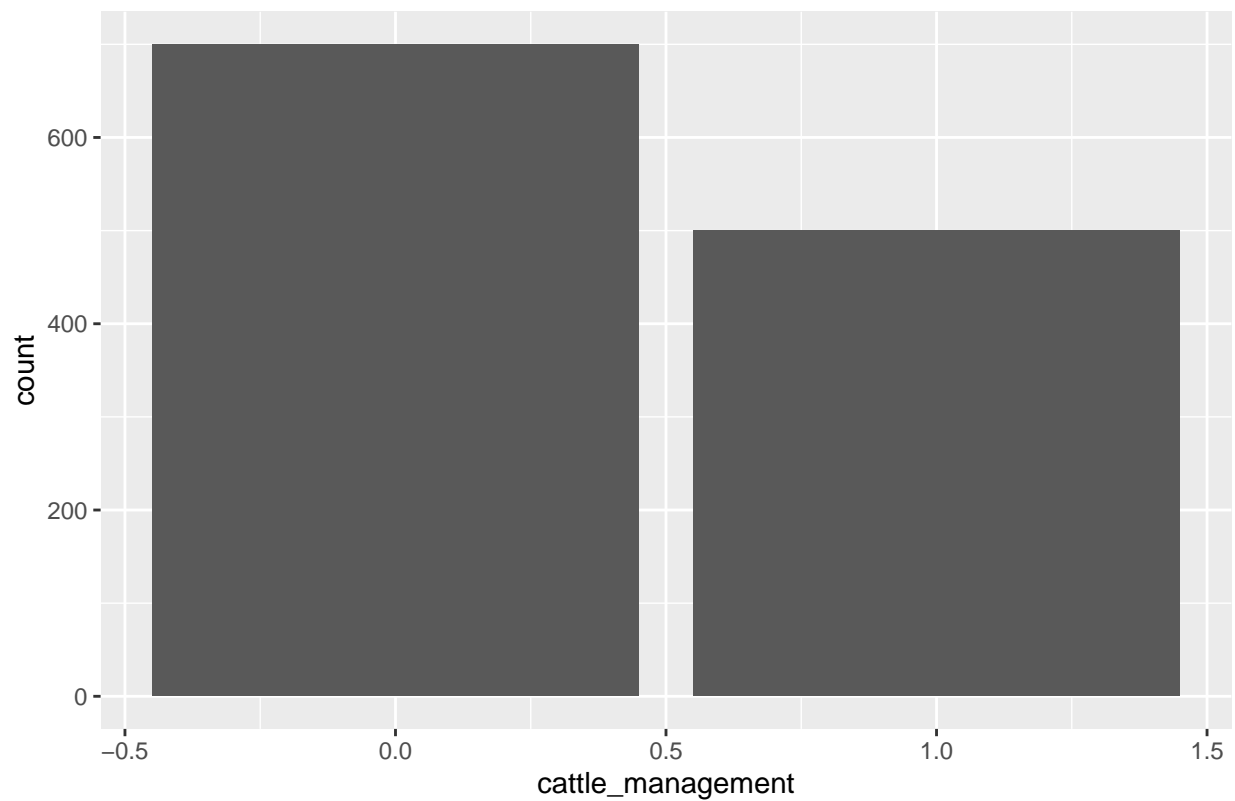
# Looking at the distribution
g <- ggplot(data = survey_final_gis_clean, aes(x = any_adaptation))
g + geom_bar() + labs(title = "Bar Plot of Any Adaptation")

```

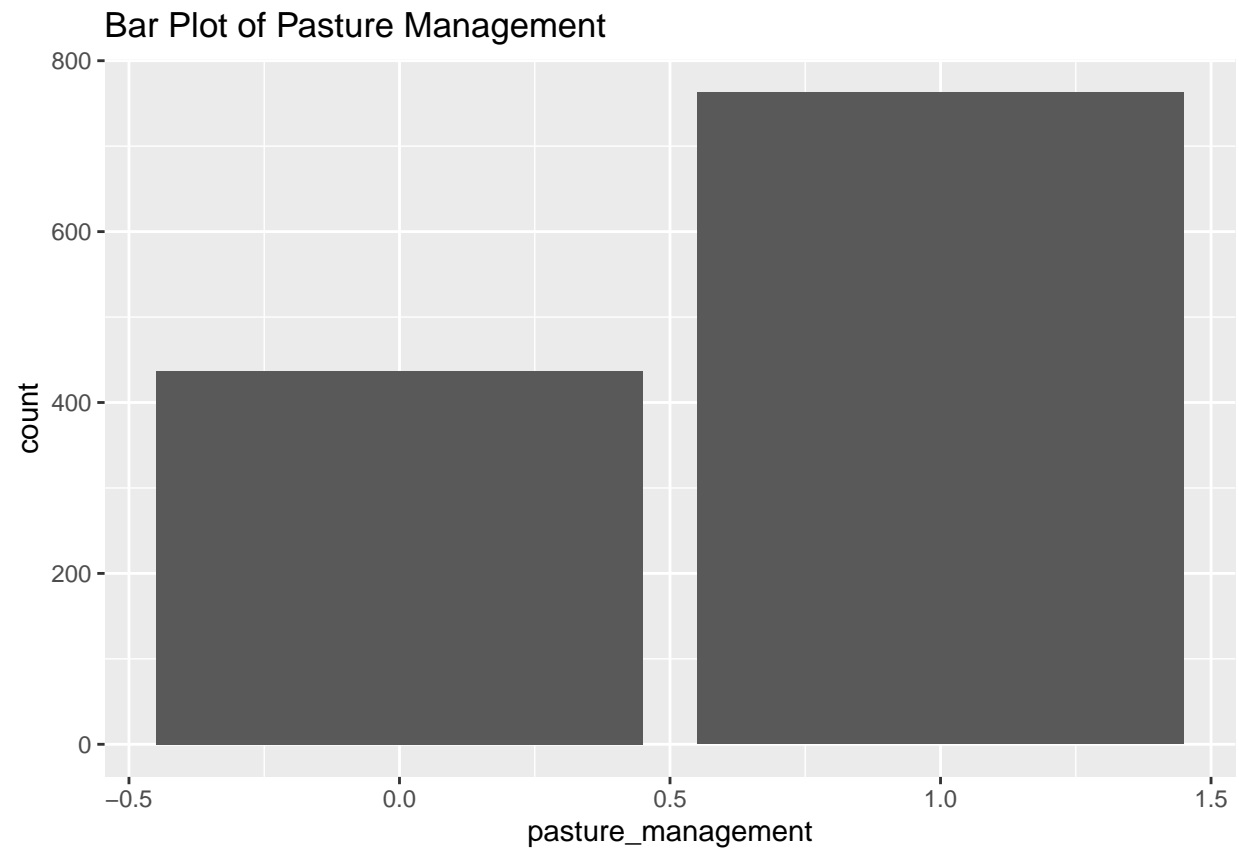


```
# Looking at the distribution for the individual variables  
g <- ggplot(data = survey_final_gis_clean, aes(x = cattle_management))  
g + geom_bar() + labs(title = "Bar Plot of Cattle Management")
```

Bar Plot of Cattle Management

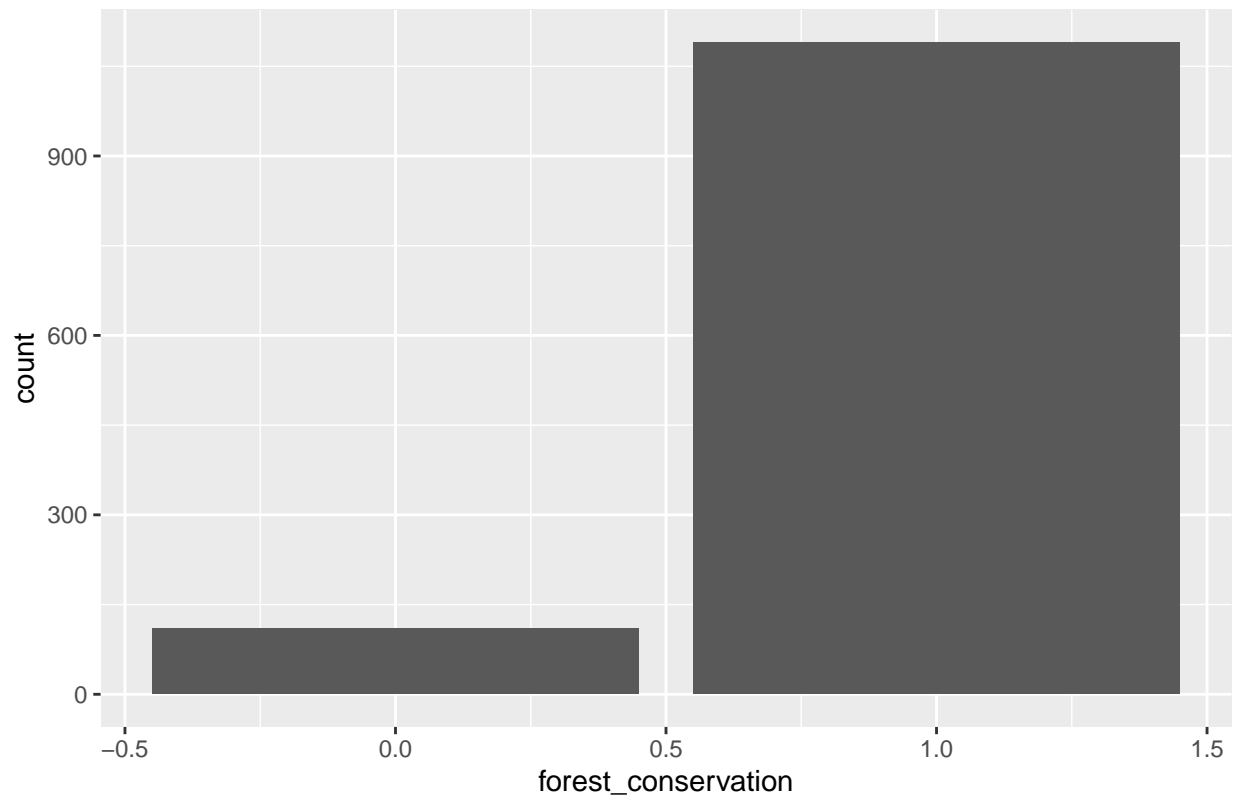


```
g <- ggplot(data = survey_final_gis_clean, aes(x = pasture_management))  
g + geom_bar() + labs(title = "Bar Plot of Pasture Management")
```

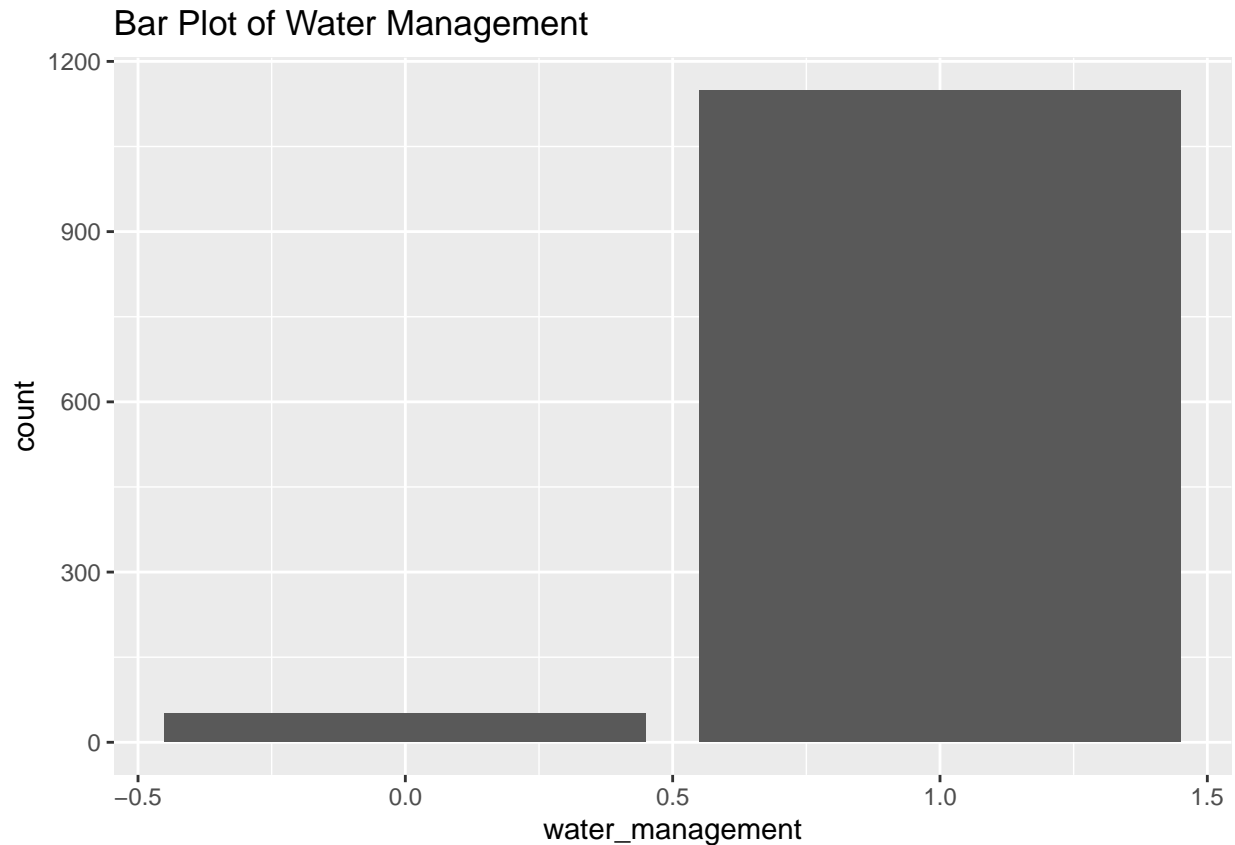



```
g <- ggplot(data = survey_final_gis_clean, aes(x = forest_conservation))  
g + geom_bar() + labs(title = "Bar Plot of Forest Conservation")
```

Bar Plot of Forest Conservation



```
g <- ggplot(data = survey_final_gis_clean, aes(x = water_management))  
g + geom_bar() + labs(title = "Bar Plot of Water Management")
```



And everyone did some sort of adaptation method. Great - this means if we drop all missing water data, we can't look at a relationship between general adaptation and results. If we include rows that are missing SPI and just use drainage area, there is still a model that can be run.

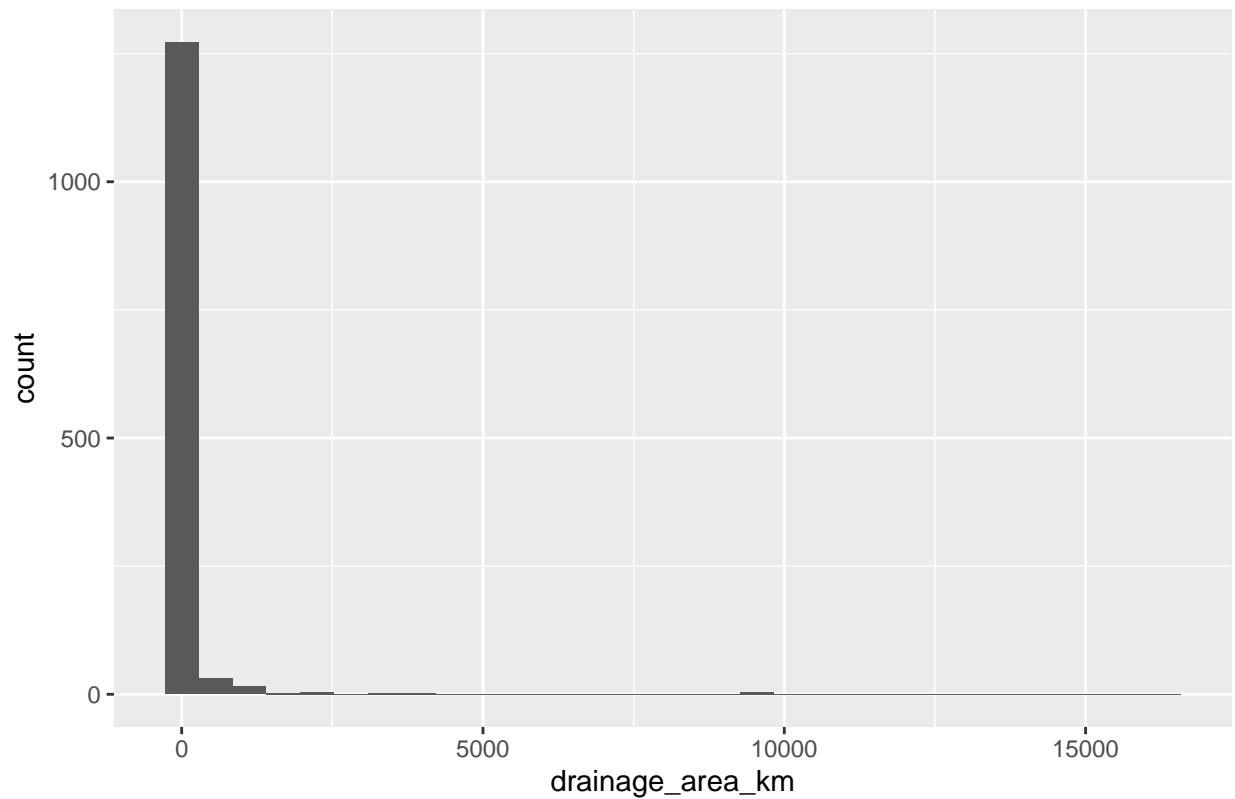
Exploratory Data Analysis

Start EDA with some individual variable histograms.

```
g <- ggplot(data = survey_final_gis, aes(x = drainage_area_km))  
g + geom_histogram() + labs(title = "Histogram of Drainage Area")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

Histogram of Drainage Area

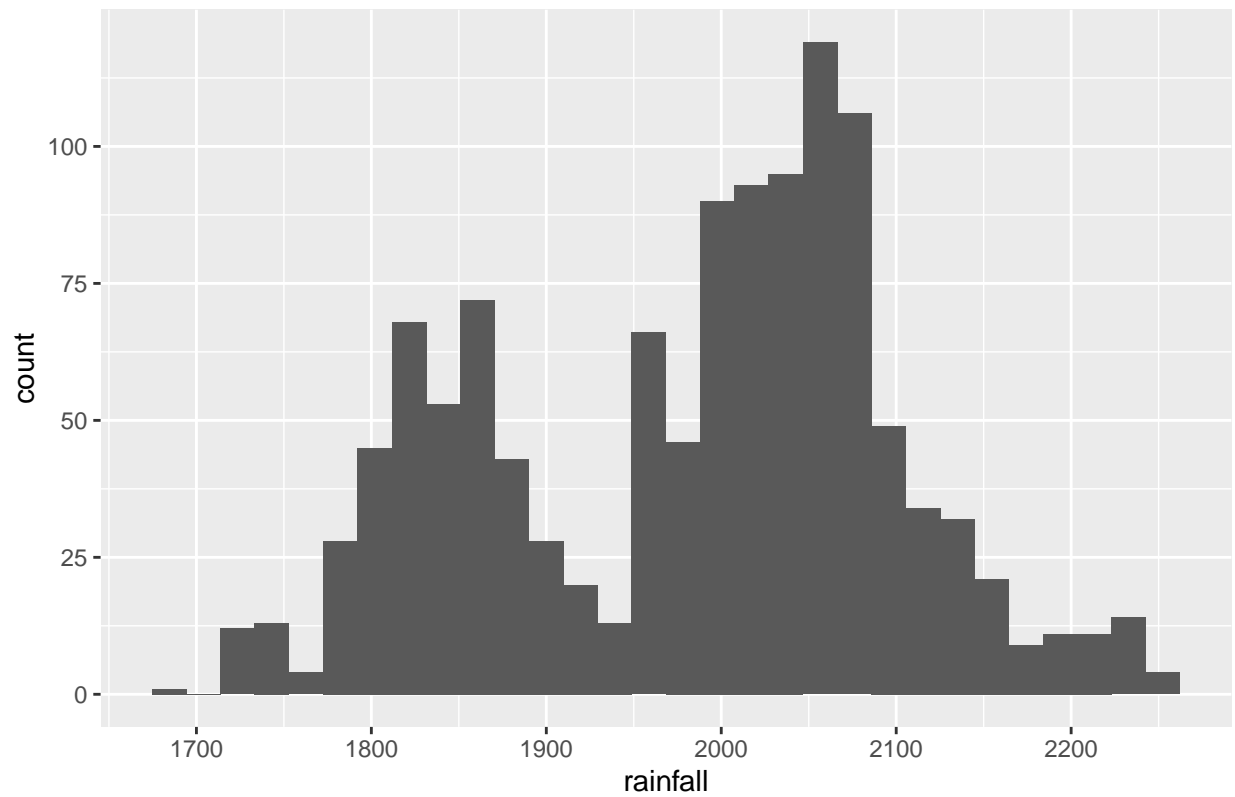


```
g <- ggplot(data = survey_final_gis, aes(x = rainfall))  
g + geom_histogram() + labs(title = "Boxplot of Rainfall")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_bin()').
```

Boxplot of Rainfall

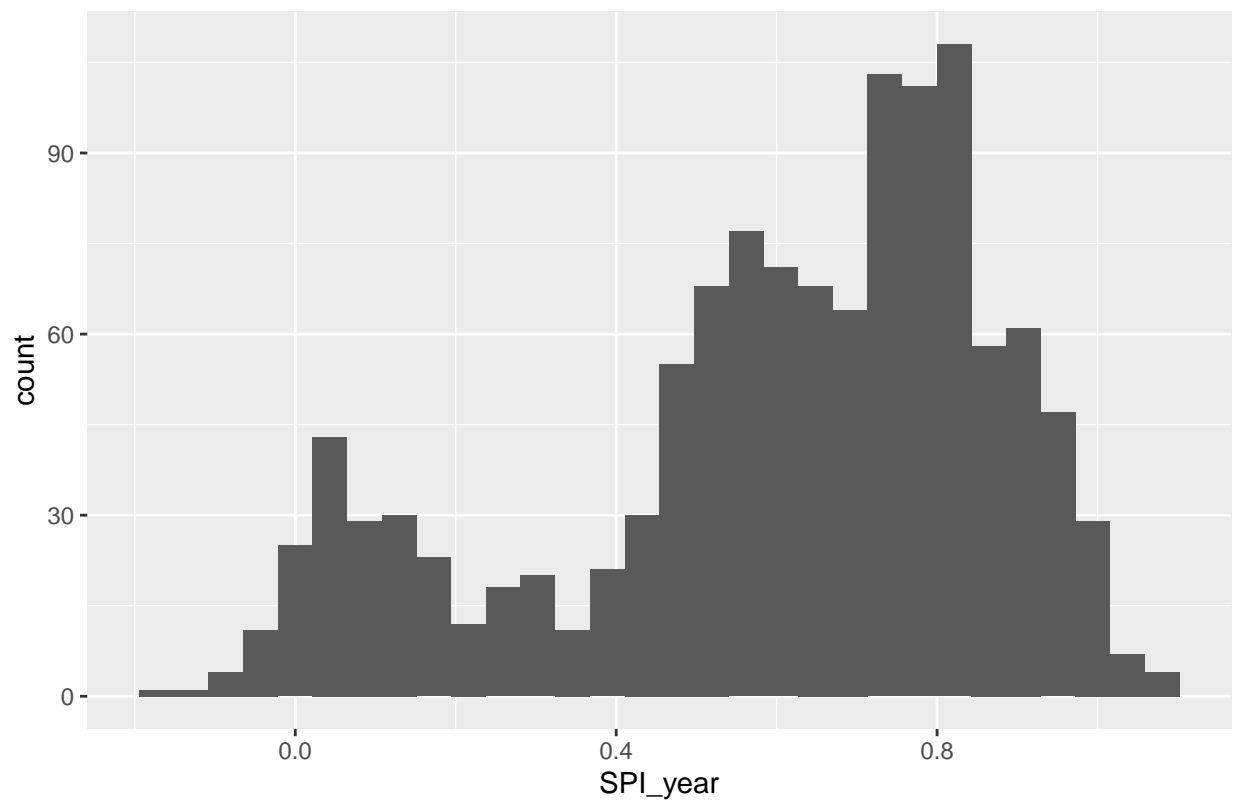


```
g <- ggplot(data = survey_final_gis, aes(x = SPI_year))
g + geom_histogram() + labs(title = "Histogram of Average Yearly SPI")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_bin()').
```

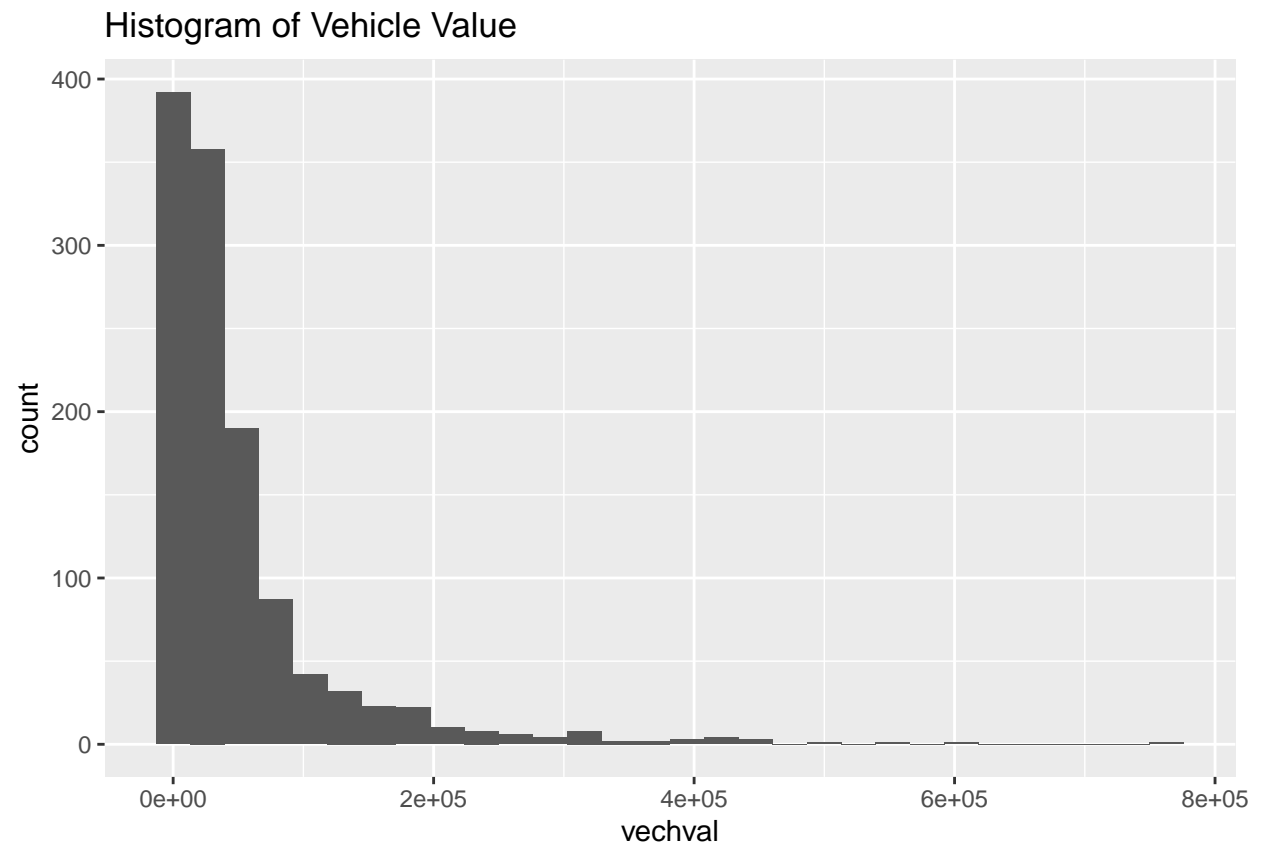
Histogram of Average Yearly SPI



```
g <- ggplot(data = survey_final_gis, aes(x = vechval))  
g + geom_histogram() + labs(title = "Histogram of Vehicle Value")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

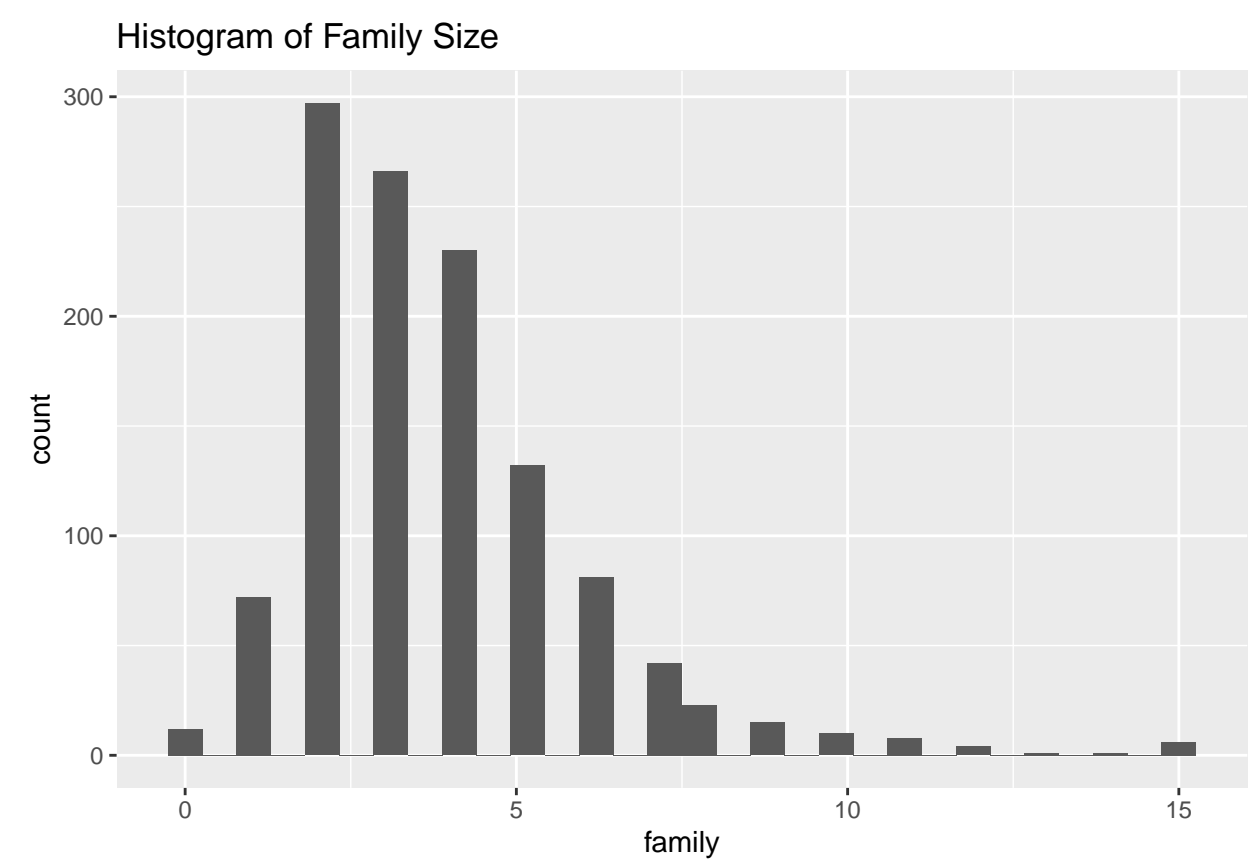
```
## Warning: Removed 140 rows containing non-finite values ('stat_bin()').
```



```
g <- ggplot(data = survey_final_gis, aes(x = family))  
g + geom_histogram() + labs(title = "Histogram of Family Size")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

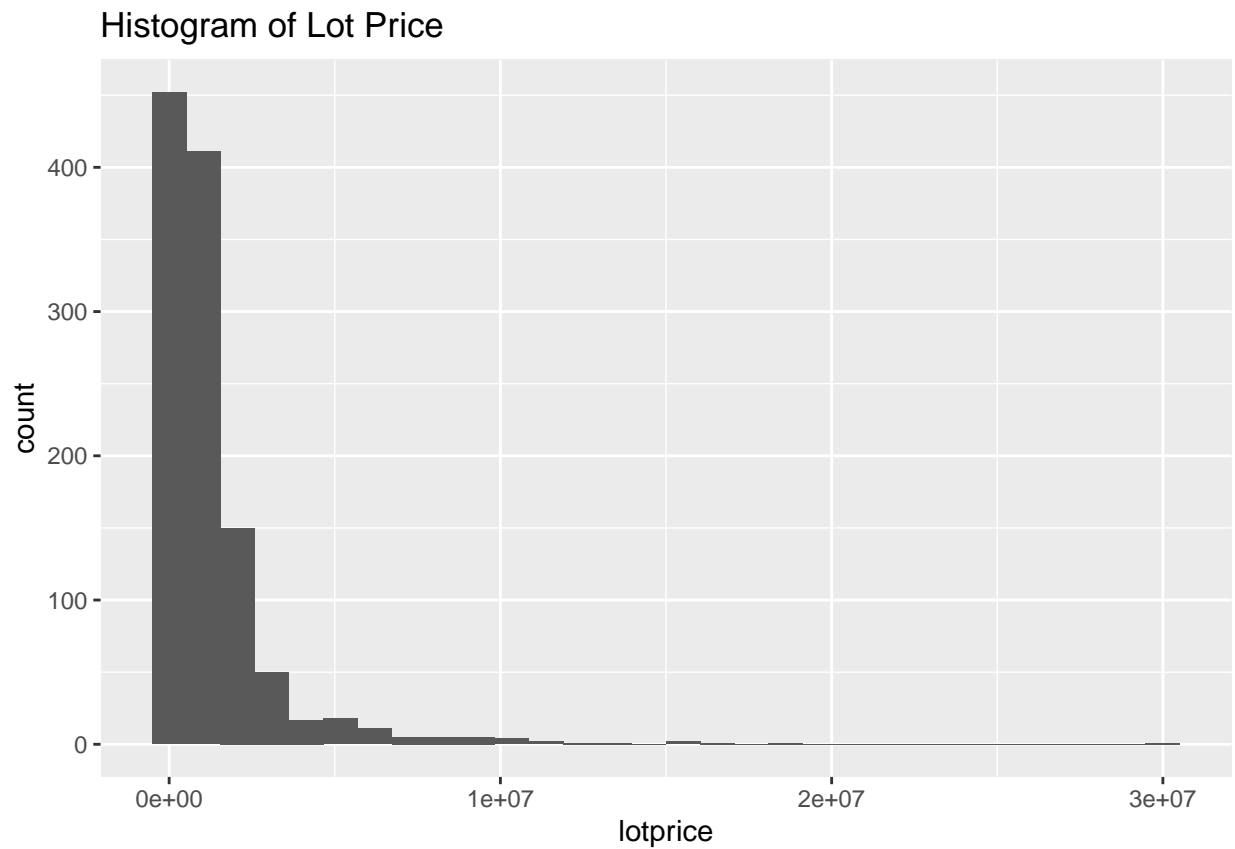
```
## Warning: Removed 140 rows containing non-finite values ('stat_bin()').
```



```
g <- ggplot(data = survey_final_gis, aes(x = lotprice))  
g + geom_histogram() + labs(title = "Histogram of Lot Price")
```

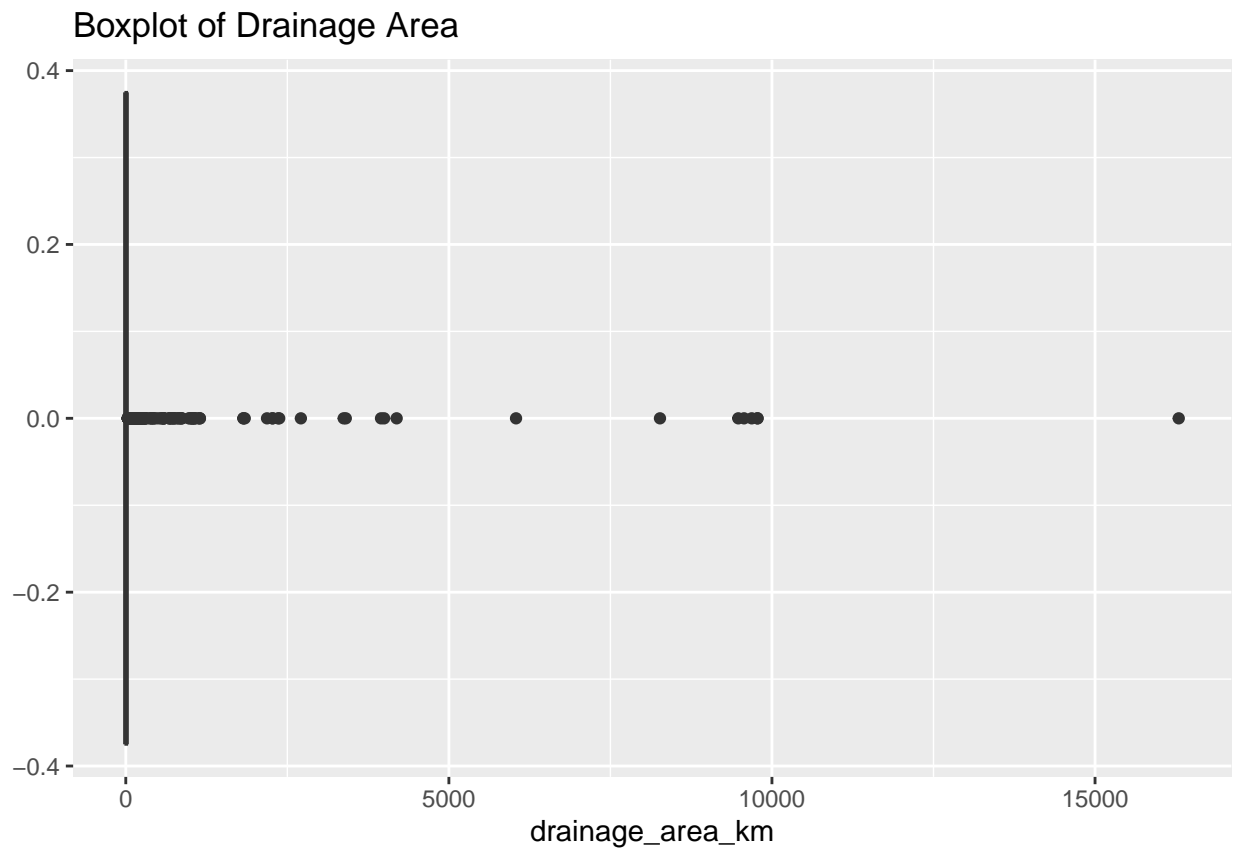
```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

```
## Warning: Removed 203 rows containing non-finite values ('stat_bin()').
```

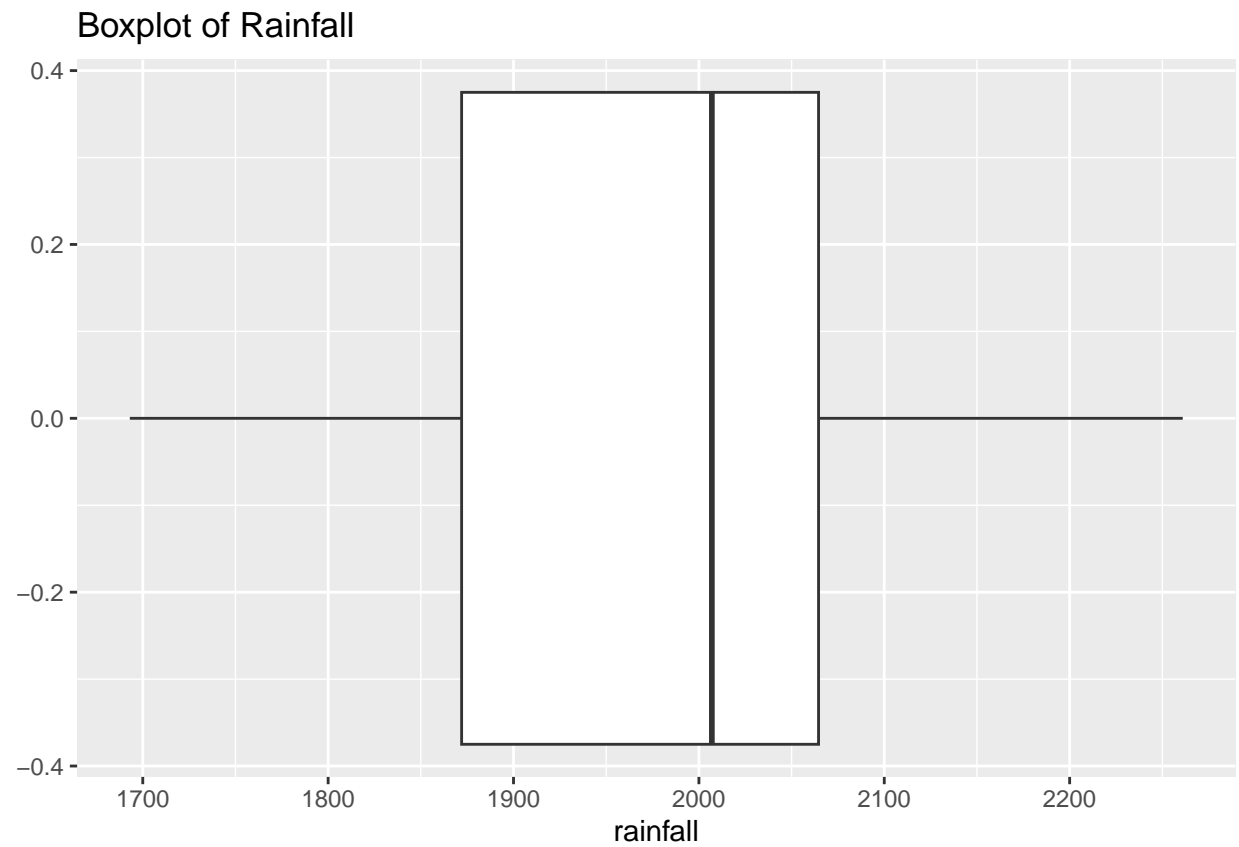
Next, we can look at some box plots

```
g <- ggplot(data = survey_final_gis, aes(x = drainage_area_km))  
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area")
```



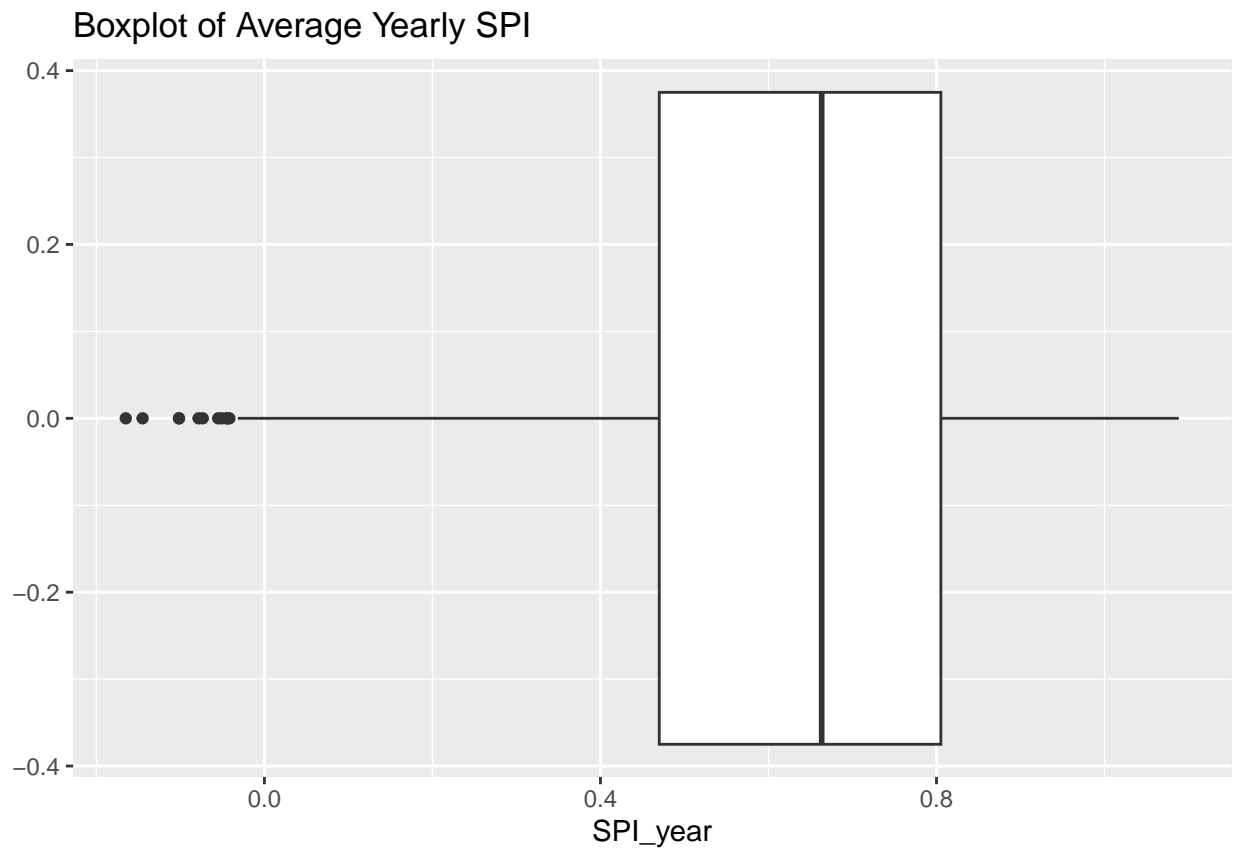
```
g <- ggplot(data = survey_final_gis, aes(x = rainfall))  
g + geom_boxplot() + labs(title = "Boxplot of Rainfall")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



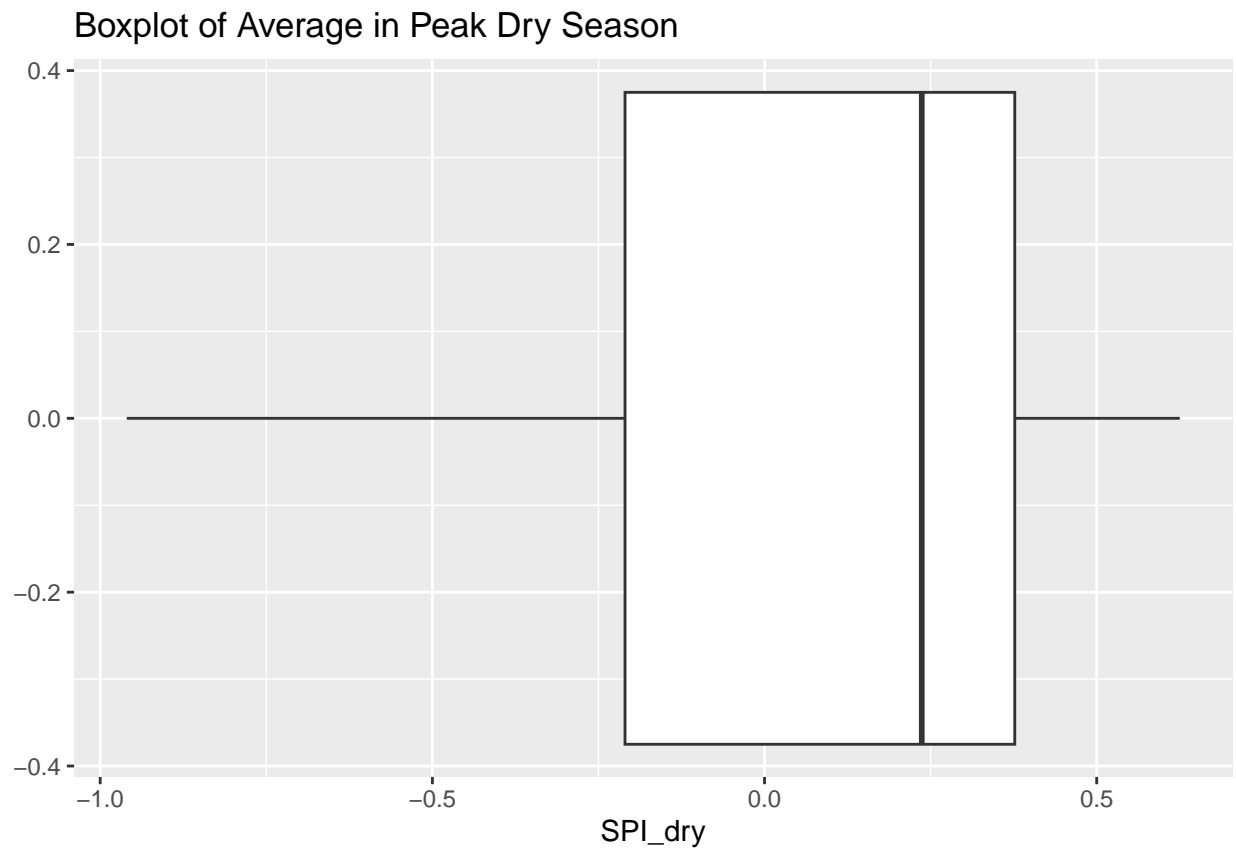
```
g <- ggplot(data = survey_final_gis, aes(x = SPI_year))  
g + geom_boxplot() + labs(title = "Boxplot of Average Yearly SPI")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



```
g <- ggplot(data = survey_final_gis, aes(x = SPI_dry))  
g + geom_boxplot() + labs(title = "Boxplot of Average in Peak Dry Season")
```

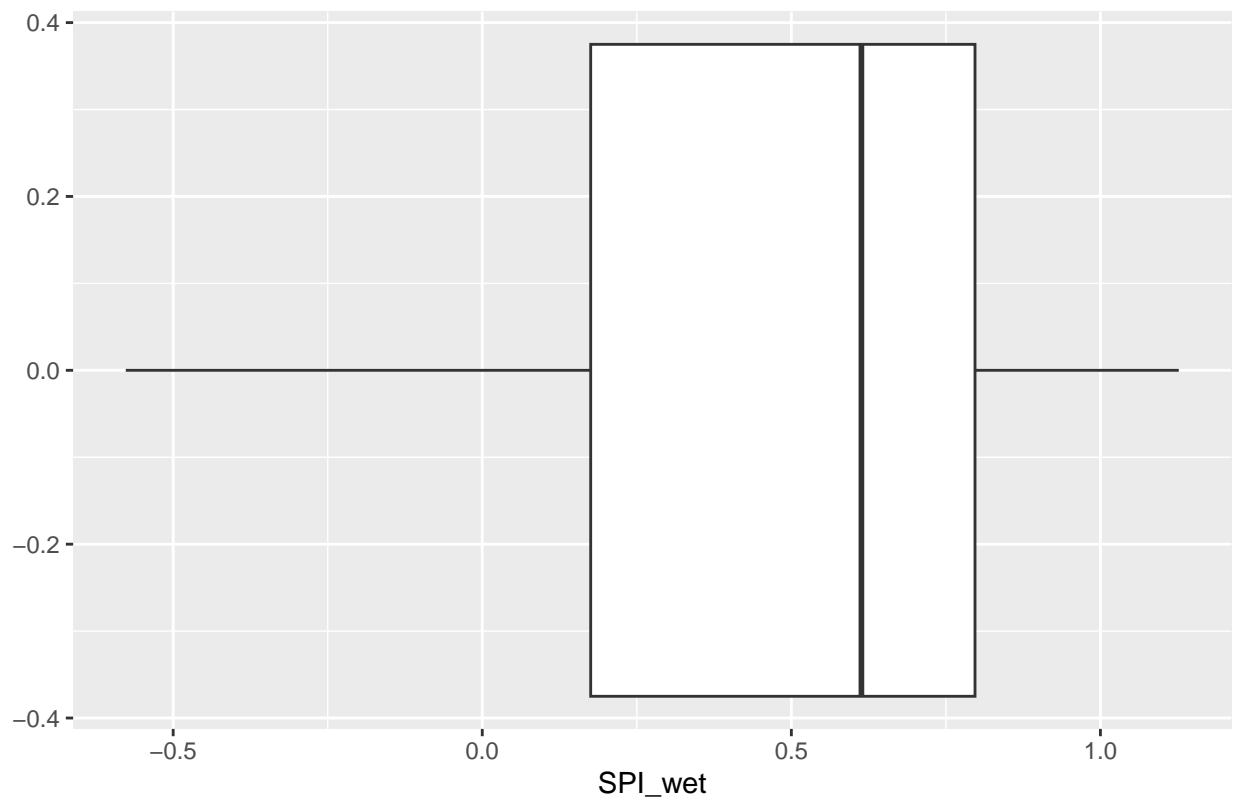
```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



```
g <- ggplot(data = survey_final_gis, aes(x = SPI_wet))  
g + geom_boxplot() + labs(title = "Boxplot of Average in Peak Wet Season")
```

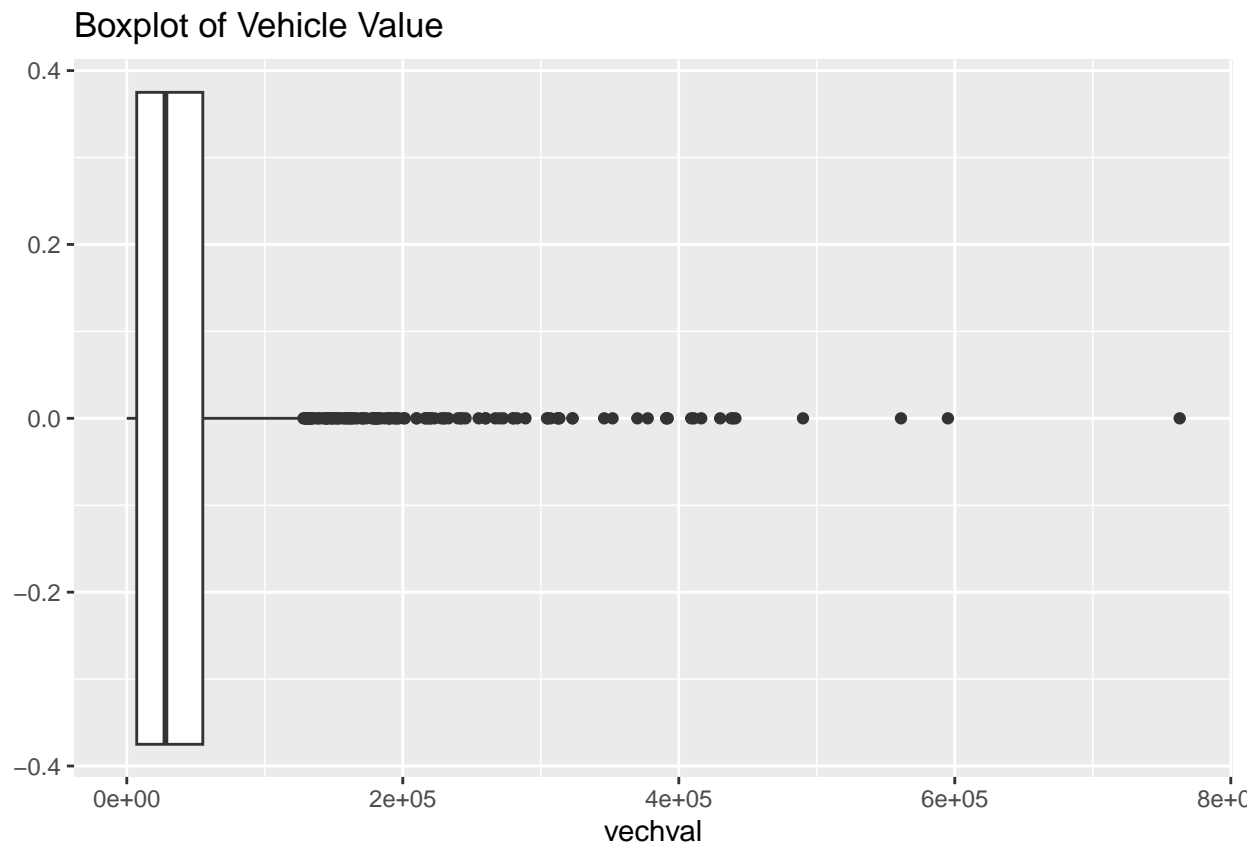
```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```

Boxplot of Average in Peak Wet Season



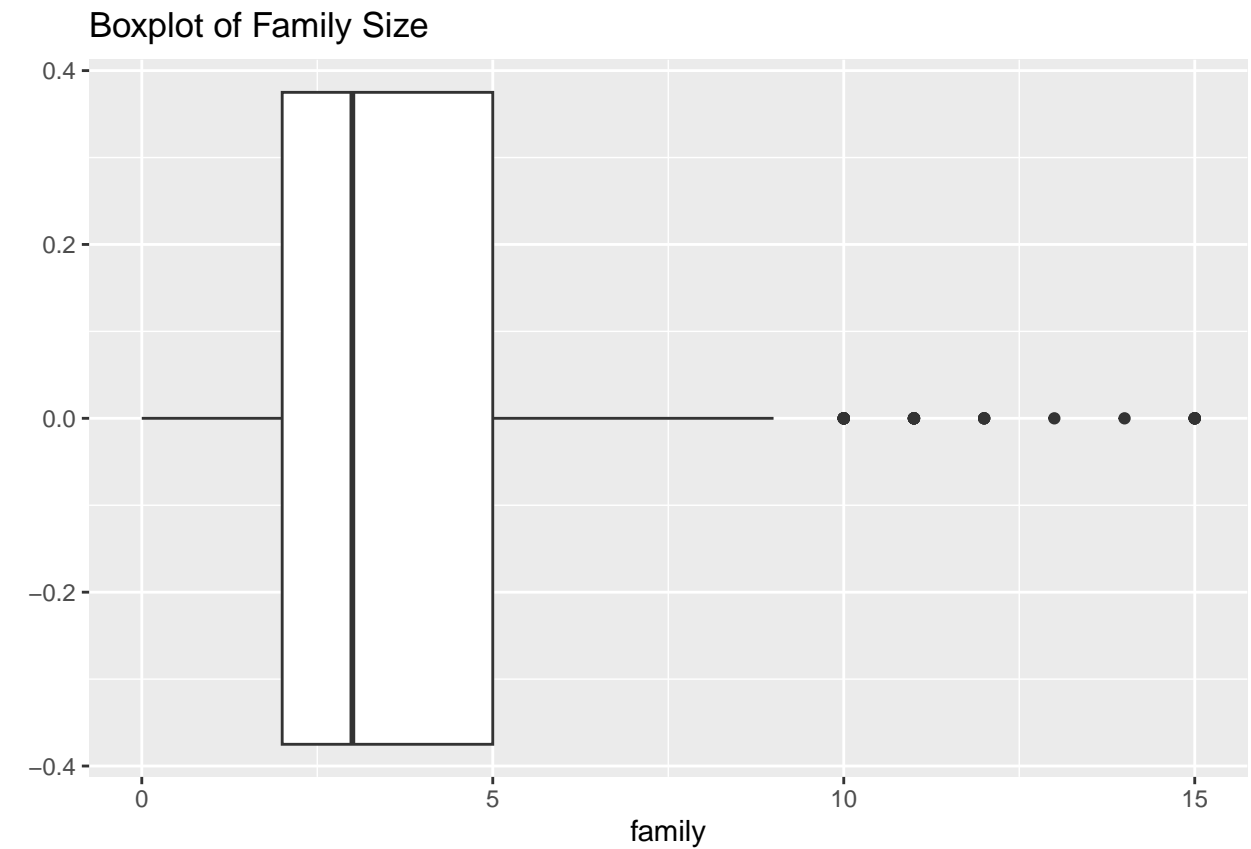
```
g <- ggplot(data = survey_final_gis, aes(x = vechval))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



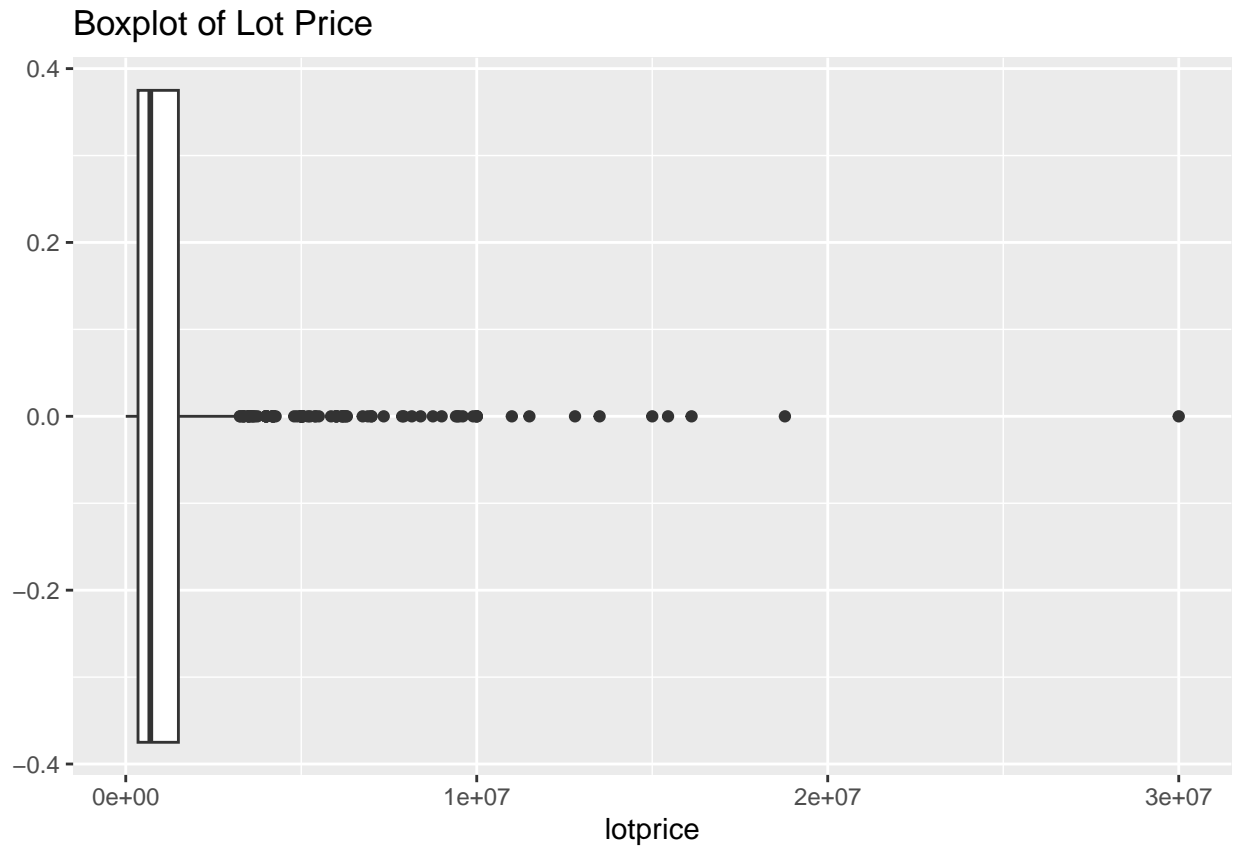
```
g <- ggplot(data = survey_final_gis, aes(x = family))  
g + geom_boxplot() + labs(title = "Boxplot of Family Size")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



```
g <- ggplot(data = survey_final_gis, aes(x = lotprice))  
g + geom_boxplot() + labs(title = "Boxplot of Lot Price")
```

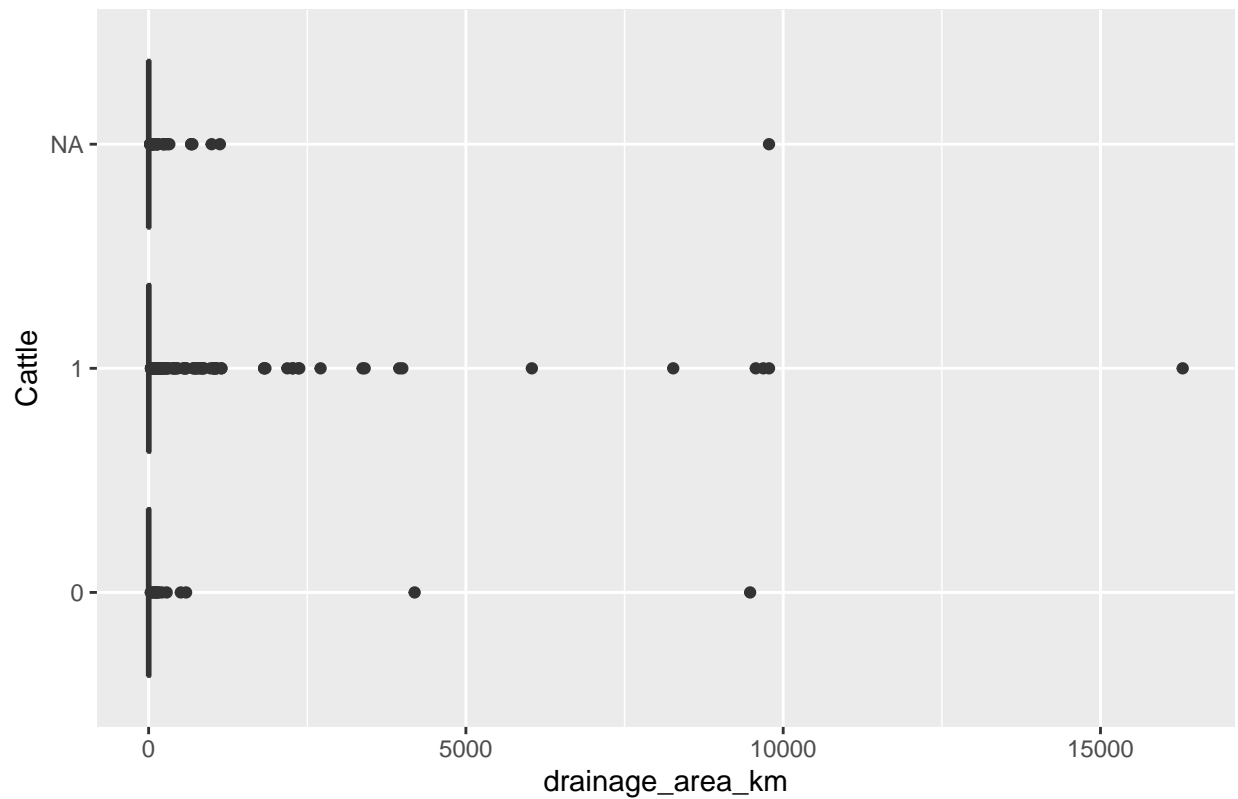
```
## Warning: Removed 203 rows containing non-finite values ('stat_boxplot()').
```

Let's look at boxplots of drainage area and vehicle value (wealth) grouped by cattle.

```
g <- ggplot(data = survey_final_gis, aes(x = drainage_area_km, y = as.factor(havecattle)))  
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by Cattle") +  
  ylab("Cattle")
```

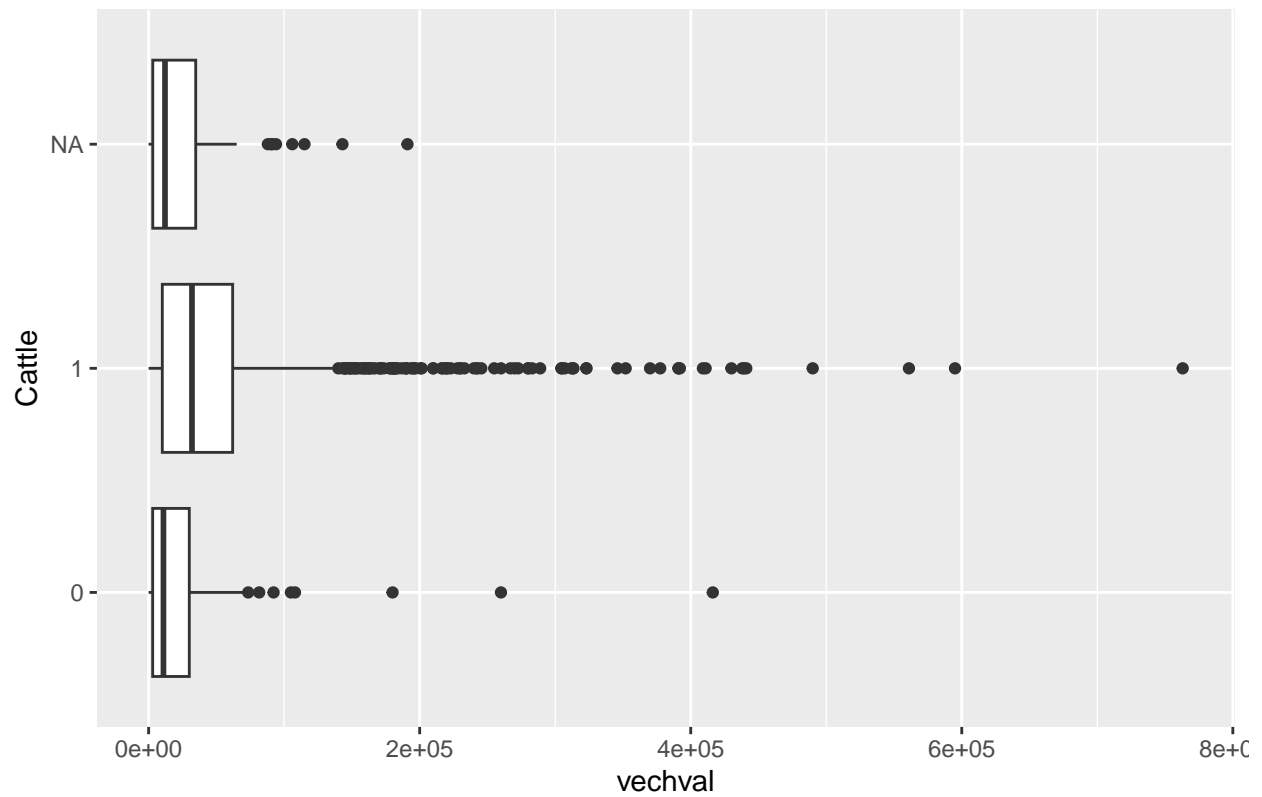
Boxplot of Drainage Area grouped by Cattle



```
g <- ggplot(data = survey_final_gis, aes(x = vechval, y = as.factor(havecattle)))
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by Cattle") +
  ylab("Cattle")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

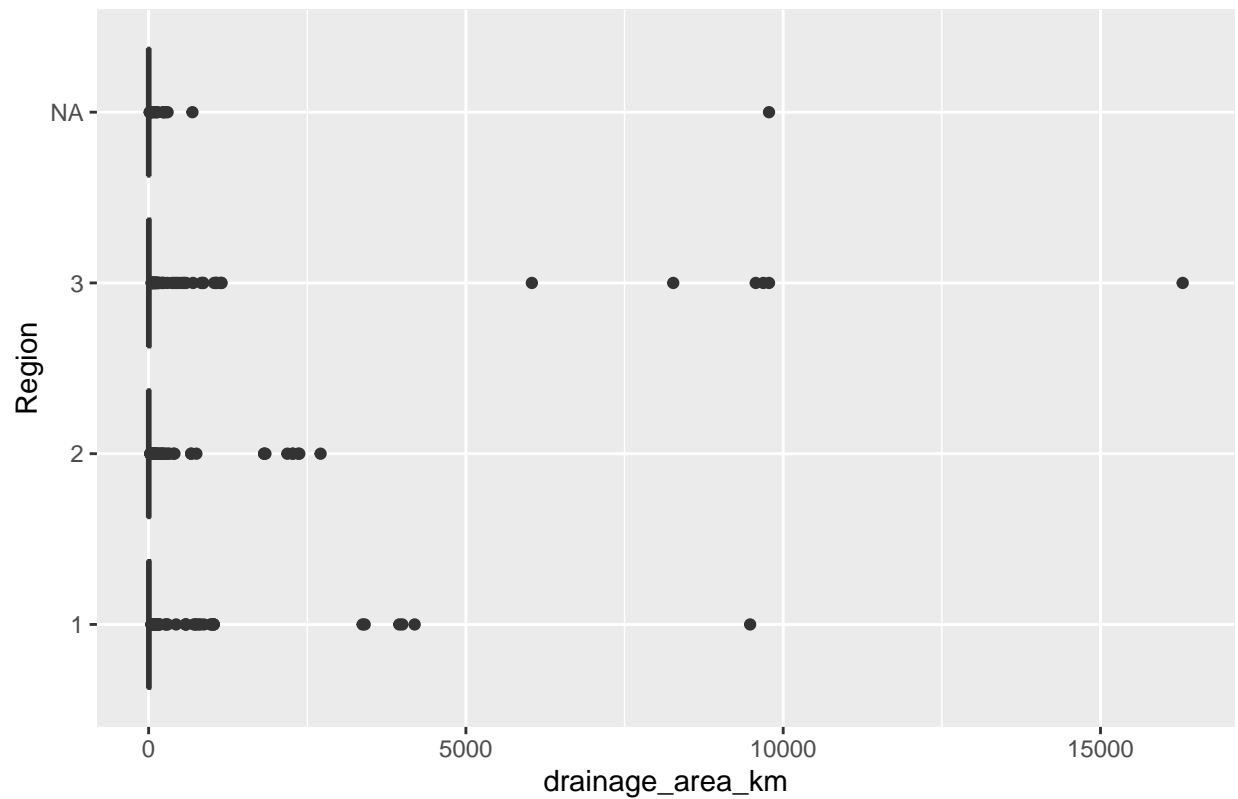
Boxplot of Vehicle Value grouped by Cattle



Let's look at a few box plots split by region

```
g <- ggplot(data = survey_final_gis, aes(x = drainage_area_km, y = as.factor(studycode)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by Region") +
  ylab("Region")
```

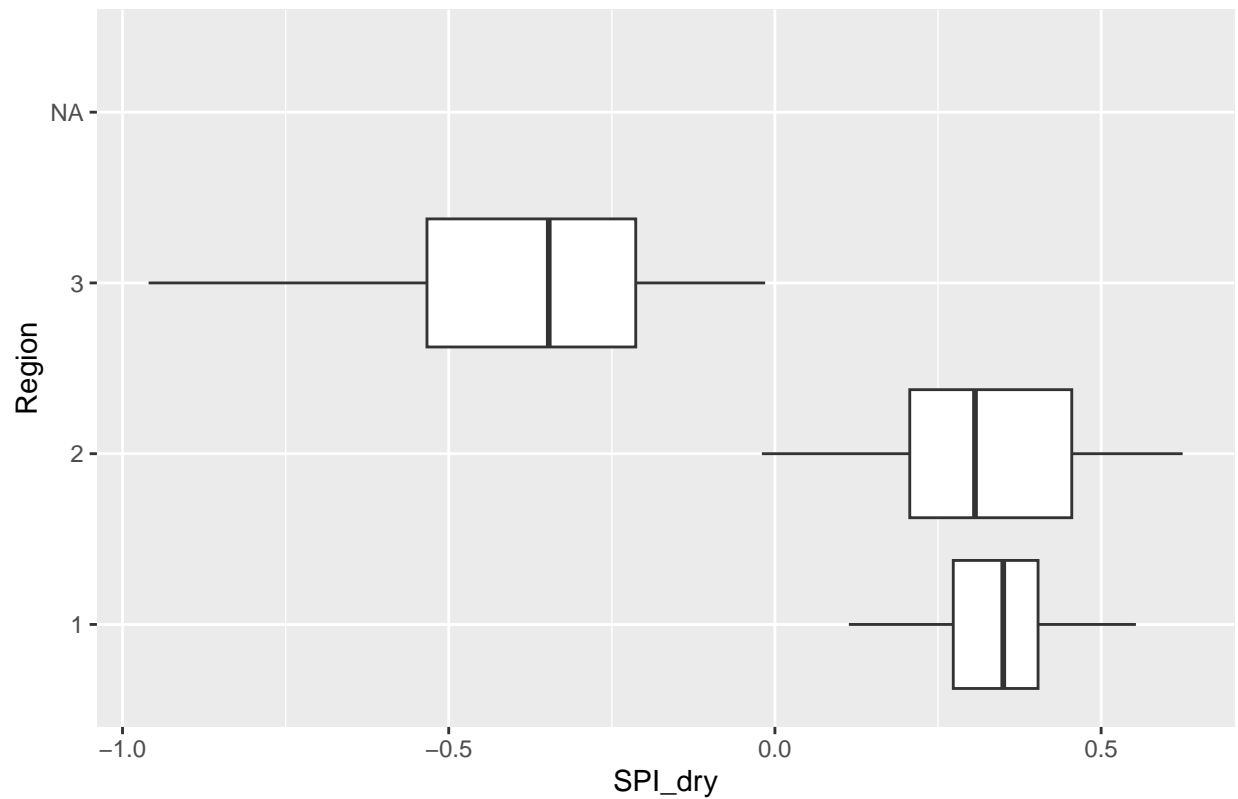
Boxplot of Drainage Area grouped by Region



```
g <- ggplot(data = survey_final_gis, aes(x = SPI_dry, y = as.factor(studycode)))
g + geom_boxplot() + labs(title = "Boxplot of Dry Season SPI grouped by Region") +
  ylab("Region")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

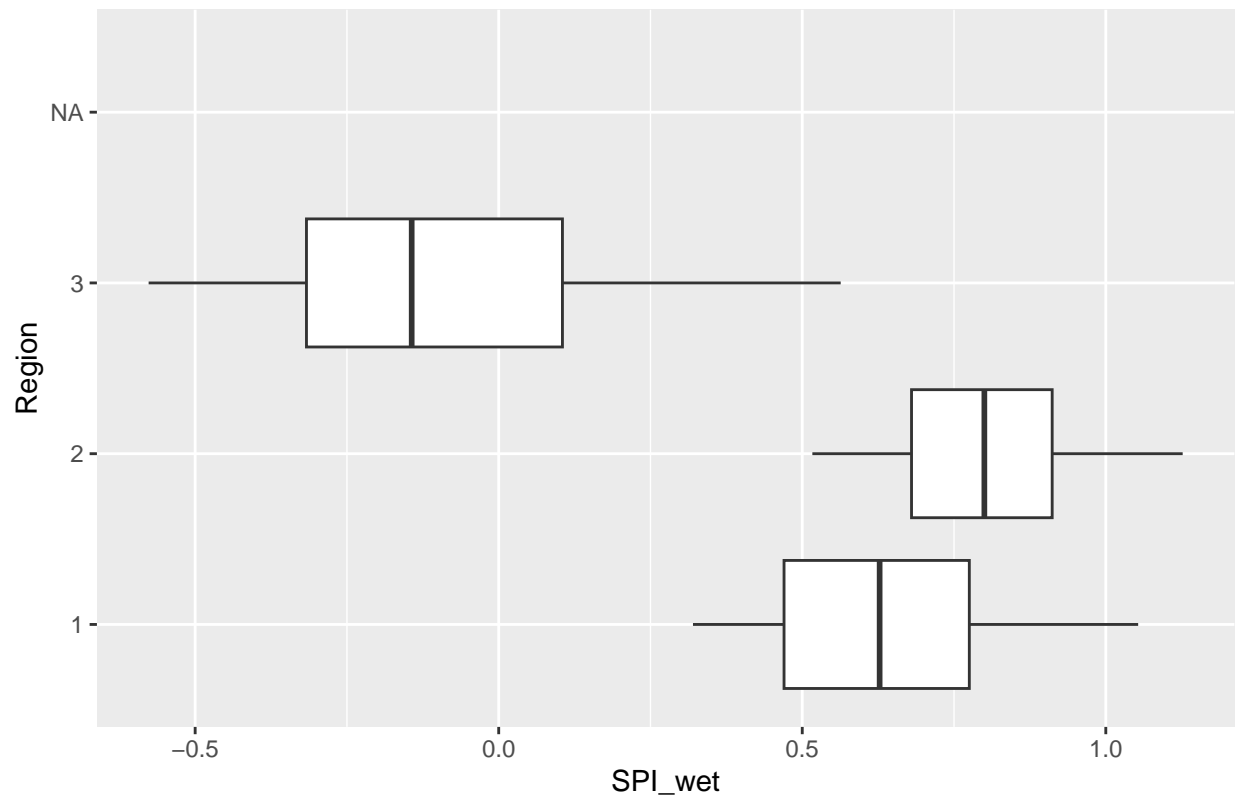
Boxplot of Dry Season SPI grouped by Region



```
g <- ggplot(data = survey_final_gis, aes(x = SPI_wet, y = as.factor(studycode)))  
g + geom_boxplot() + labs(title = "Boxplot of Wet Season SPI grouped by Region") +  
  ylab("Region")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

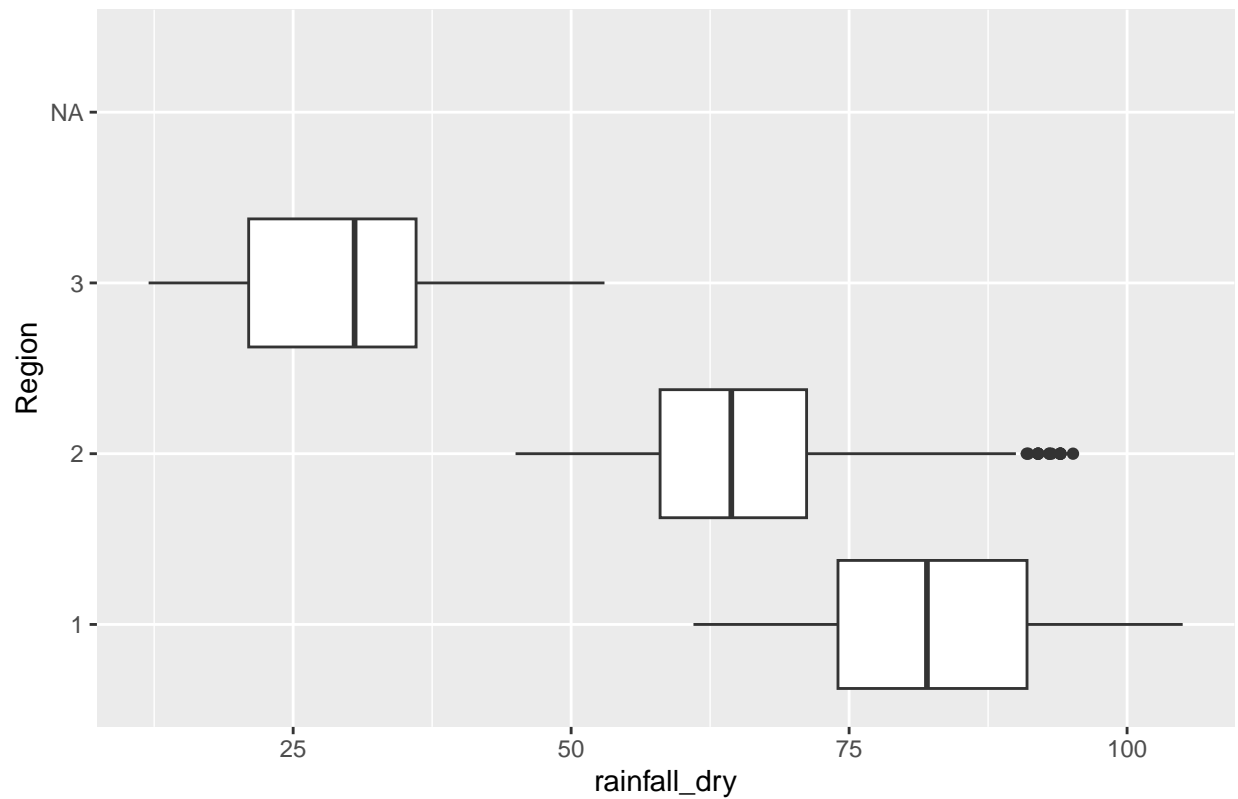
Boxplot of Wet Season SPI grouped by Region



```
g <- ggplot(data = survey_final_gis, aes(x = rainfall_dry, y = as.factor(studycode)))  
g + geom_boxplot() + labs(title = "Boxplot of Dry Season Rainfall grouped by Region") +  
  ylab("Region")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

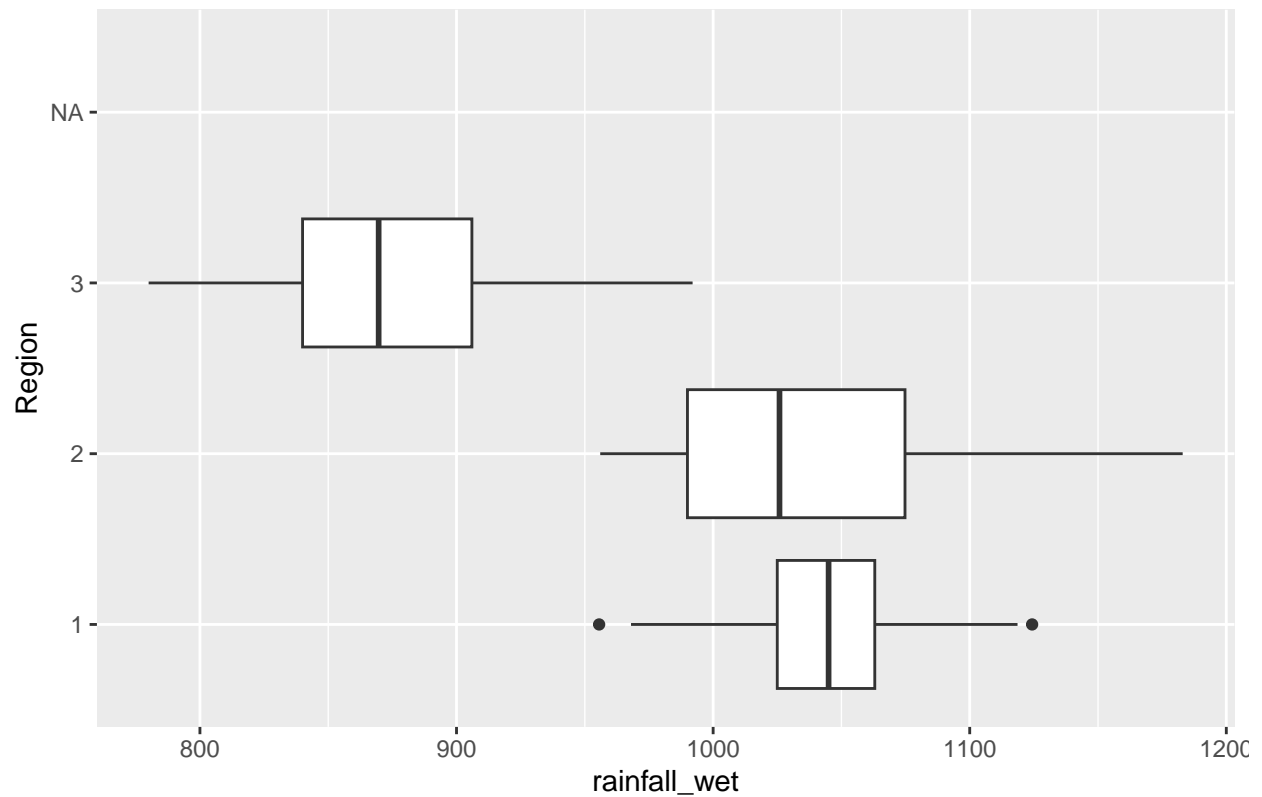
Boxplot of Dry Season Rainfall grouped by Region



```
g <- ggplot(data = survey_final_gis, aes(x = rainfall_wet, y = as.factor(studycode)))  
g + geom_boxplot() + labs(title = "Boxplot of Wet Season Rainfall grouped by Region") +  
  ylab("Region")
```

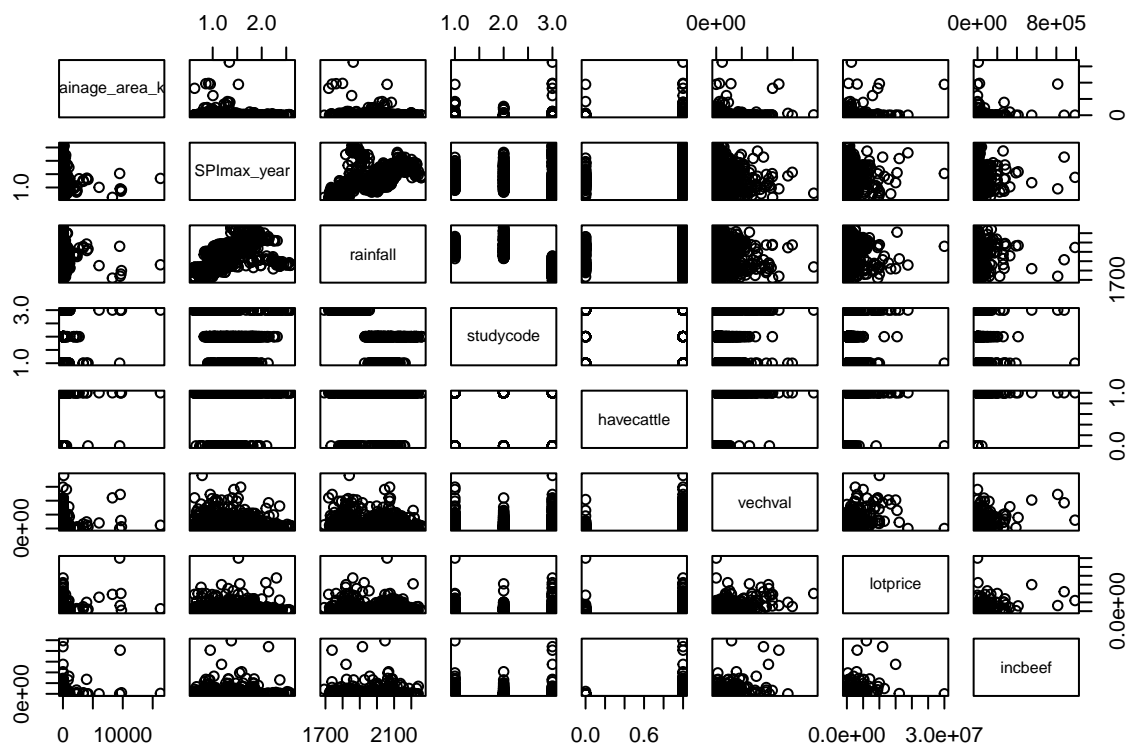
```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```

Boxplot of Wet Season Rainfall grouped by Region



Next lets look at pair plots

```
pairs_subset <- survey_final_gis %>% select(c(drainage_area_km, SPImax_year, rainfall, studycode, havec))
pairs(pairs_subset)
```

Let's look at how many households fall into each region

```
table(survey_final_gis$studycode)
```

```
##
##    1    2    3
## 368 441 391
```

Let's also look at some basic summary statistics

```
describe(survey_final_gis[c('drainage_area_km', 'rainfall', 'rainfall_wet', 'rainfall_dry', 'SPI_wet',
```

```
##
##      vars    n      mean      sd    min      max
## drainage_area_km    1 1340    126.10    846.22    0.01 16295.36
## rainfall            2 1200   1982.59   116.73 1693.00   2261.00
## rainfall_wet        3 1200    984.51    89.15  780.00   1183.00
## rainfall_dry        4 1200    59.03    24.22   12.00   105.00
## SPI_wet             5 1200     0.46     0.44  -0.58     1.13
## SPI_dry             6 1200     0.10     0.37  -0.96     0.62
## incbeef            7  698  27979.62  78088.34    0.00 990000.00
## vechval            8 1200   50514.90  74844.85    0.00 763000.00
## lotprice           9 1137 1315067.21 2034041.25 1000.00 30000000.00
## family            10 1200     3.75     2.18    0.00    15.00
##
##           range      se
## drainage_area_km 16295.35 23.12
```

```
## rainfall          568.00    3.37
## rainfall_wet      403.00    2.57
## rainfall_dry       93.00    0.70
## SPI_wet           1.70    0.01
## SPI_dry           1.58    0.01
## incbeef          990000.00 2955.69
## vechval           763000.00 2160.58
## lotprice          29999000.00 60322.53
## family            15.00    0.06
```

I'm curious about how many farms made multiple adaptations

```
table(survey_final_gis$cattle_management, survey_final_gis$pasture_management)
```

```
##
##      0    1
##  0 467 373
##  1 110 390
```

```
table(survey_final_gis$cattle_management, survey_final_gis$forest_conservation)
```

```
##
##      0    1
##  0 216 624
##  1   34 466
```

```
table(survey_final_gis$cattle_management, survey_final_gis$water_management)
```

```
##
##      0    1
##  0 169 671
##  1   22 478
```

```
table(survey_final_gis$pasture_management, survey_final_gis$forest_conservation)
```

```
##
##      0    1
##  0 194 383
##  1   56 707
```

```
table(survey_final_gis$pasture_management, survey_final_gis$water_management)
```

```
##
##      0    1
##  0 162 415
##  1   29 734
```

```
table(survey_final_gis$forest_conservation, survey_final_gis$water_management)
```

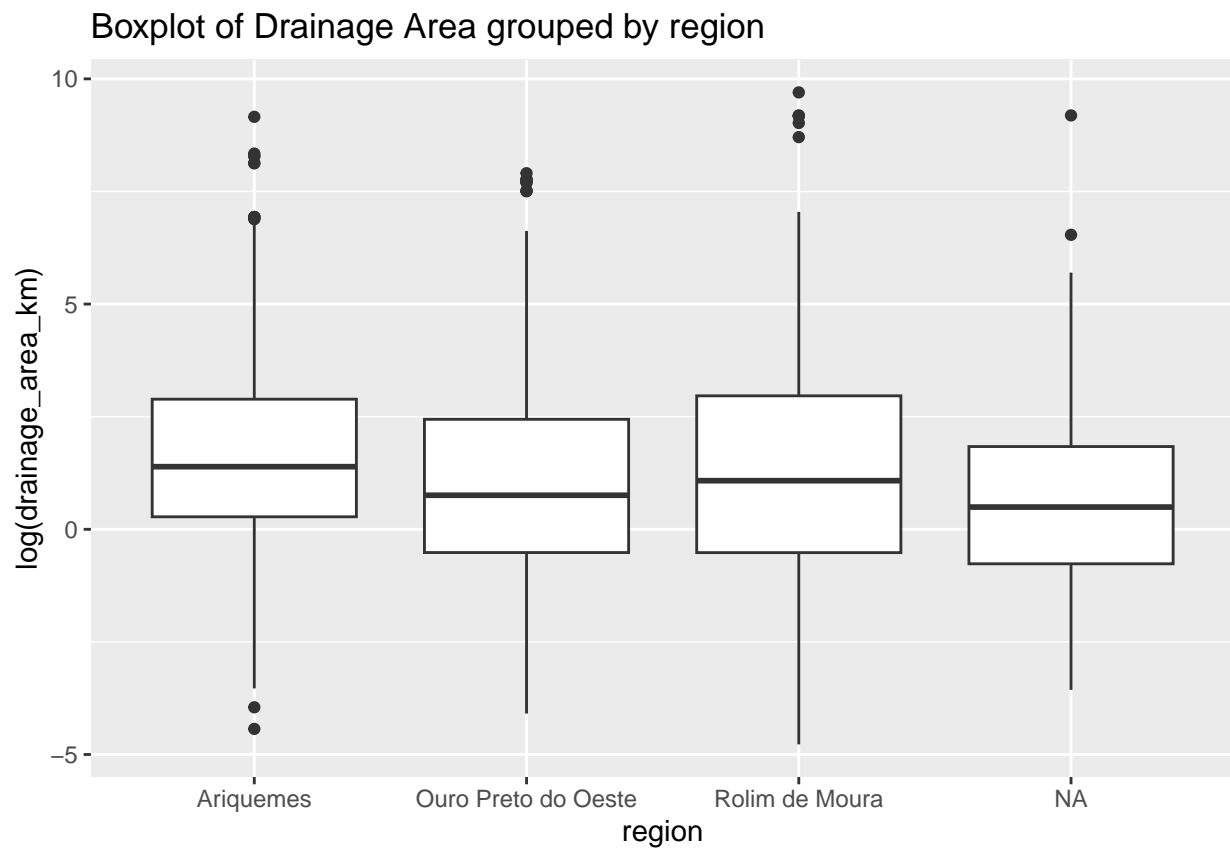
```
##
##      0      1
##    0 149 101
##    1  42 1048
```

It looks like the majority of farmers who adapted did more than one thing.

boxplot supplement

Effect of region on drainage area, rainfall and vechval

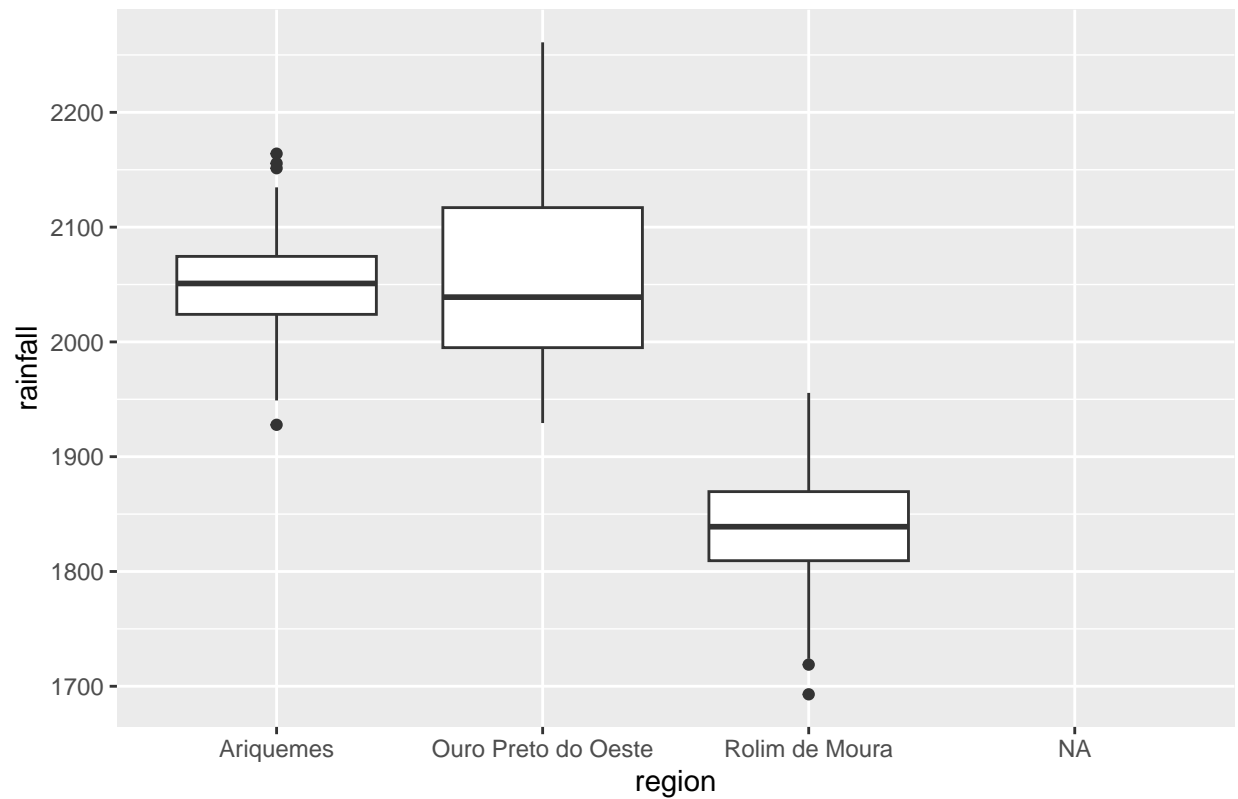
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(region)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by region") +
  xlab("region")
```



```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x= as.factor(region)))
g + geom_boxplot() + labs(title = "Boxplot of rainfall grouped by region") +
  xlab("region")
```

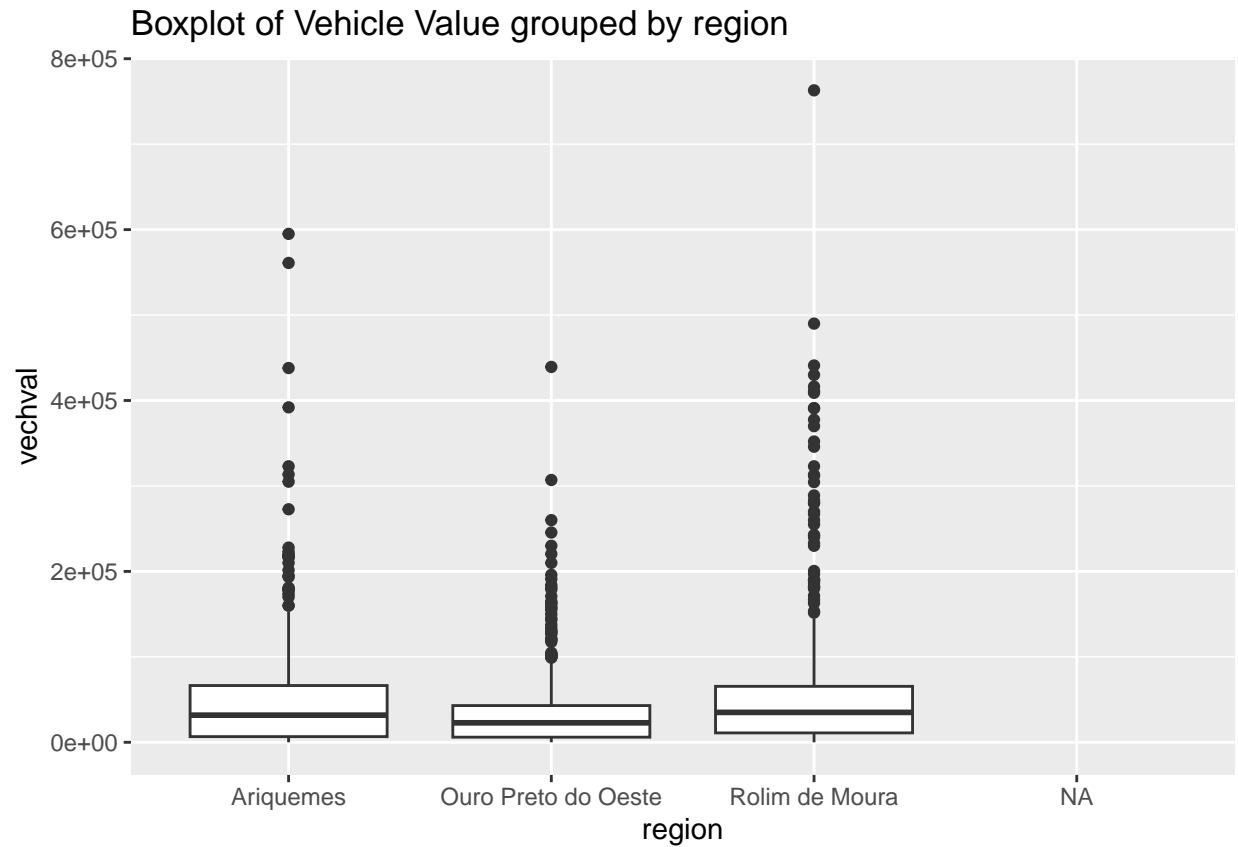
```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```

Boxplot of rainfall grouped by region



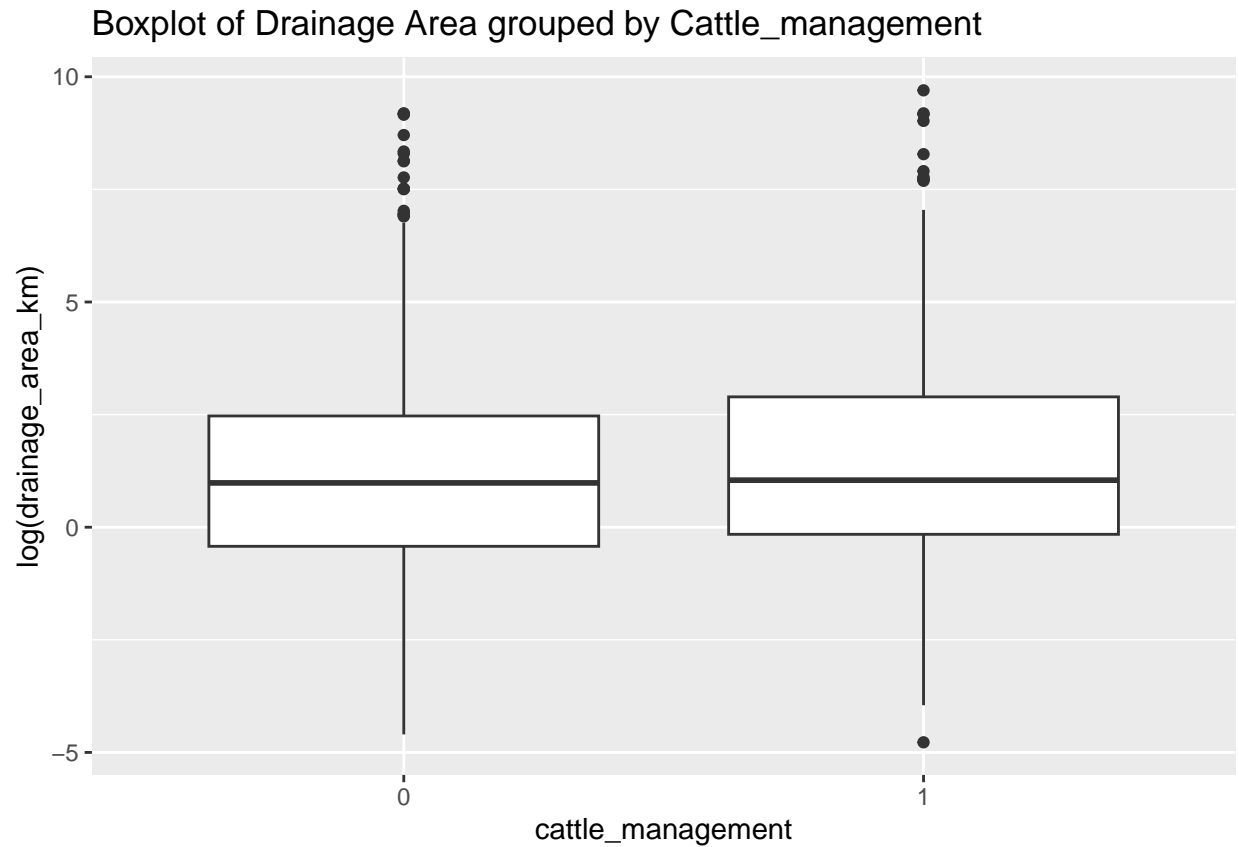
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(region)))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by region") +  
  xlab("region")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```



Effect of cattle_management on drainage area, rainfall and vechval

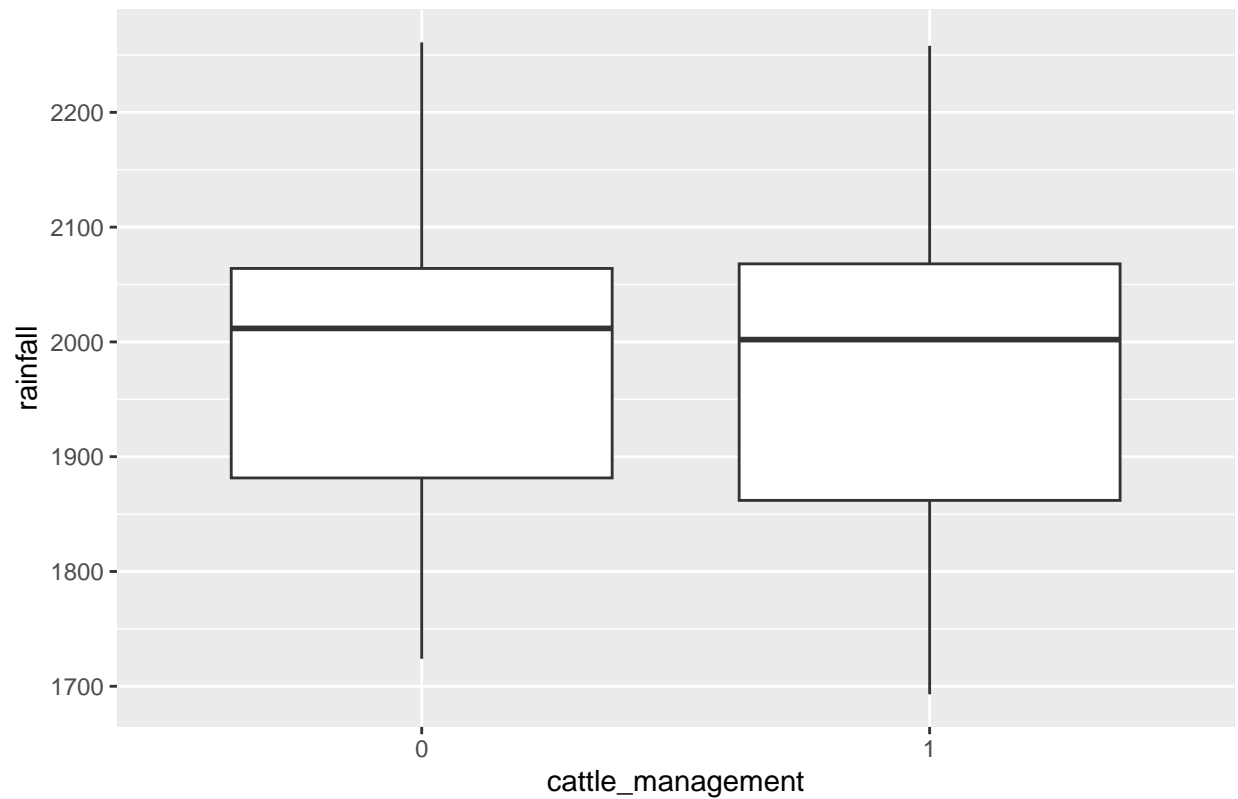
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(cattle_management)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by Cattle_management") +
  xlab("cattle_management")
```



```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x = as.factor(cattle_management)))  
g + geom_boxplot() + labs(title = "Boxplot of rainfall grouped by Cattle_management") +  
  xlab("cattle_management")
```

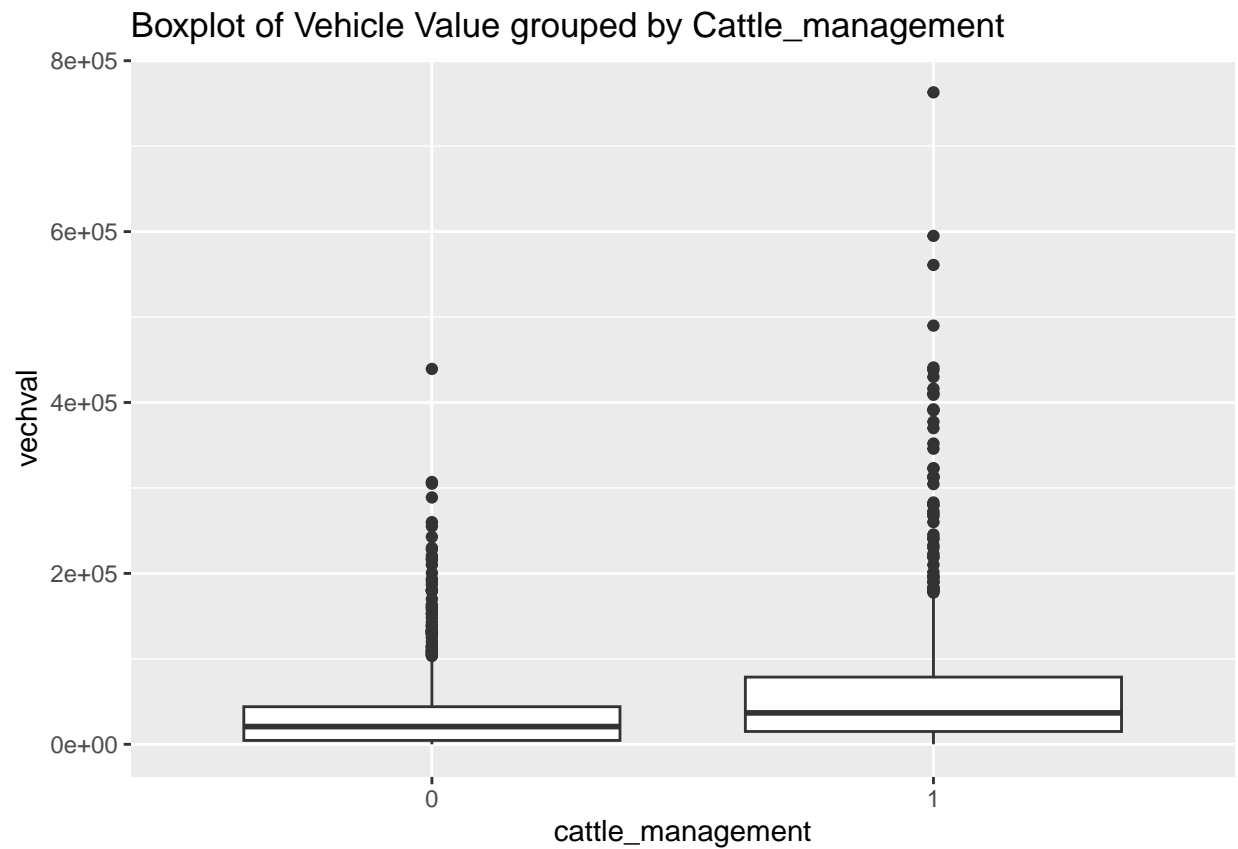
Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

Boxplot of rainfall grouped by Cattle_management



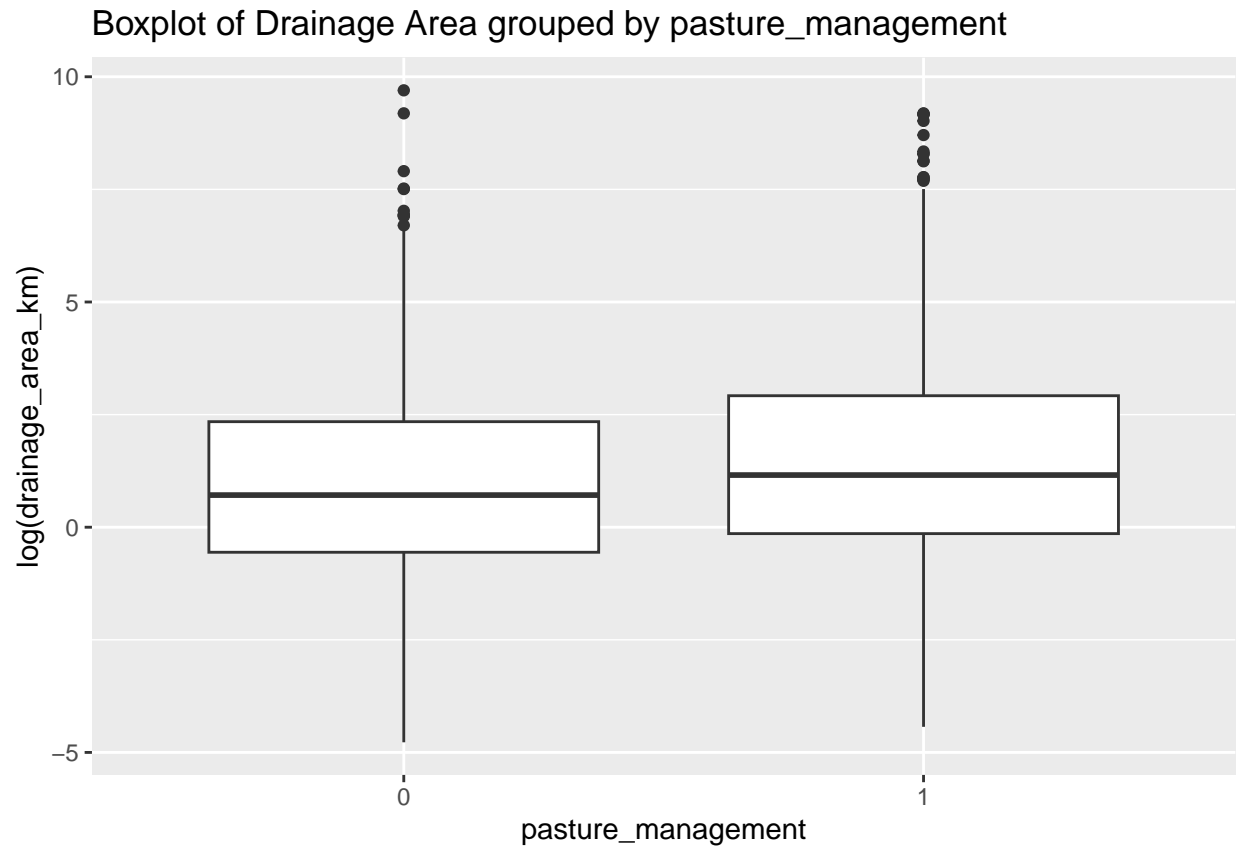
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(cattle_management)))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by Cattle_management") +  
  xlab("cattle_management")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').



Effect of pasture_management on drainage area, rainfall and vechval

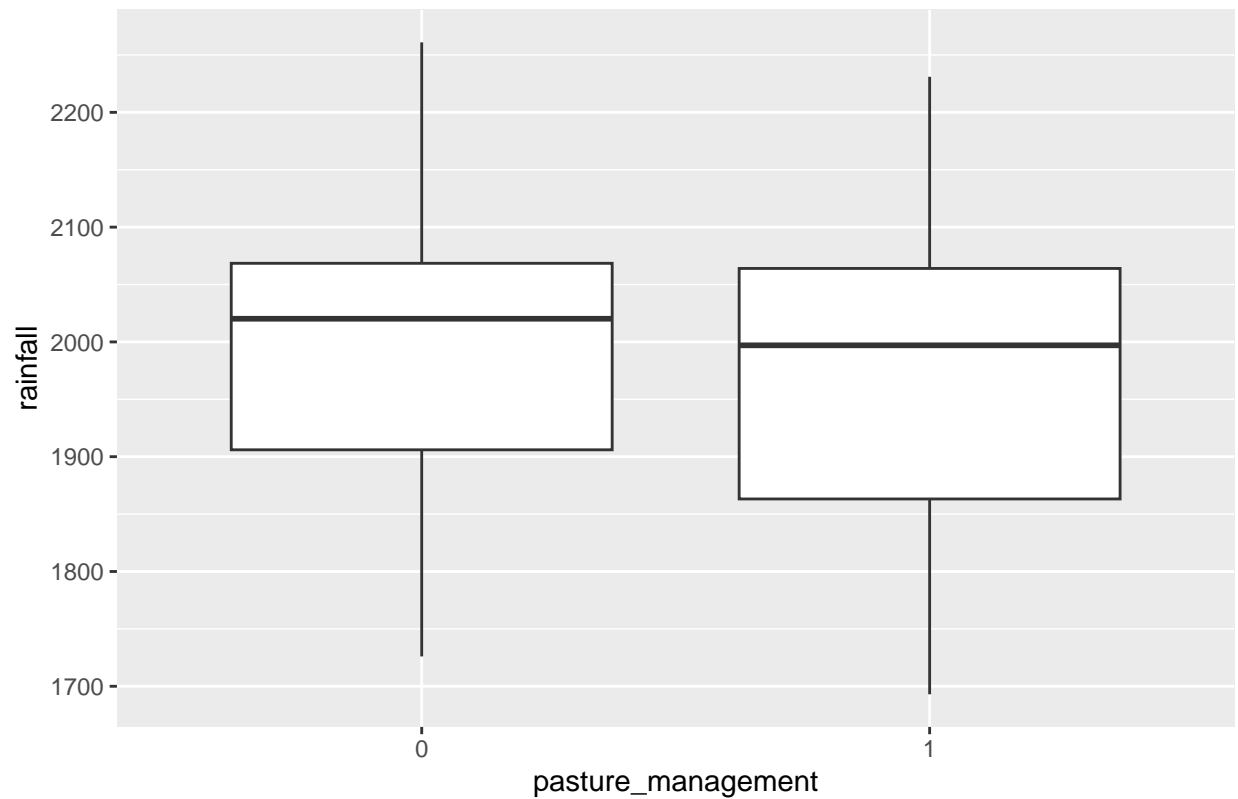
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(pasture_management)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by pasture_management") +
  xlab("pasture_management")
```

```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x = as.factor(pasture_management)))  
g + geom_boxplot() + labs(title = "Boxplot of Vrainfall grouped by pasture_management") +  
  xlab("pasture_management")
```

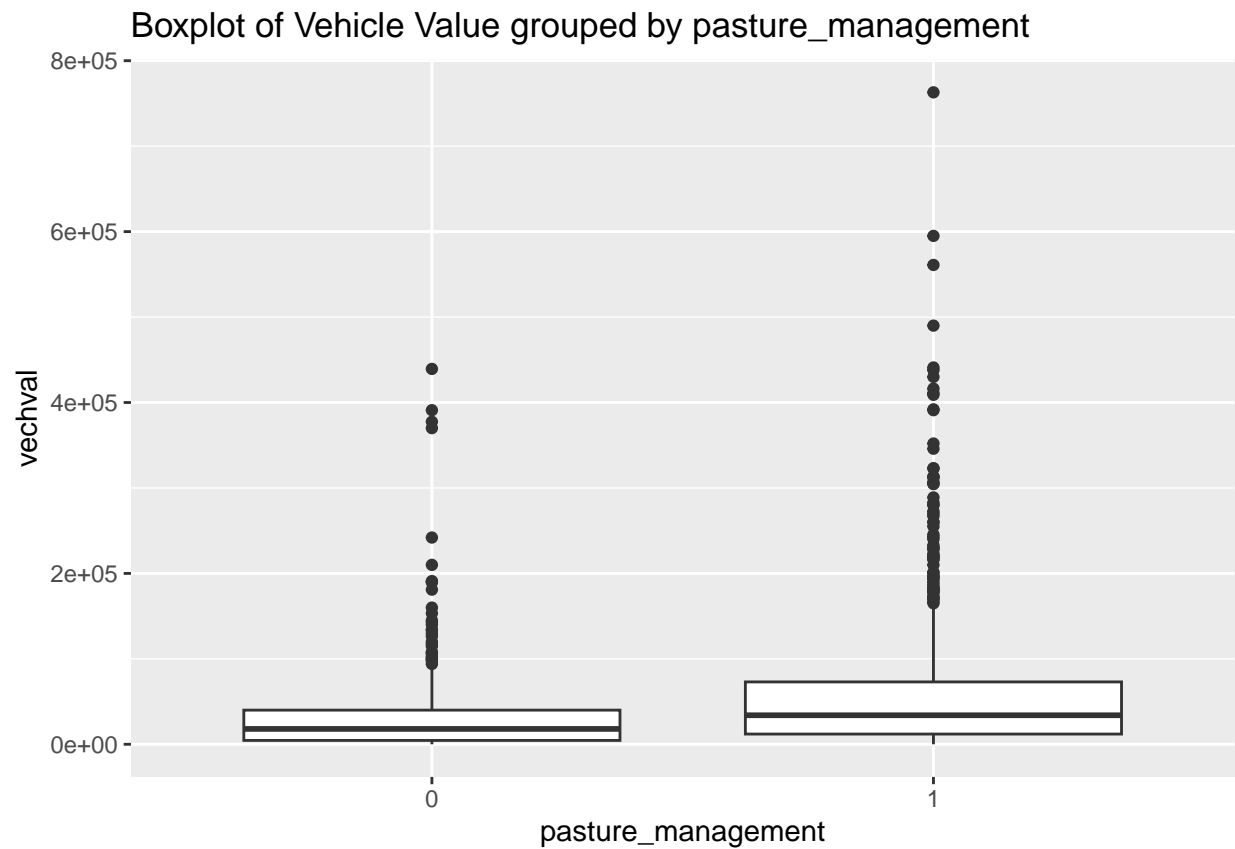
Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

Boxplot of Vrainfall grouped by pasture_management



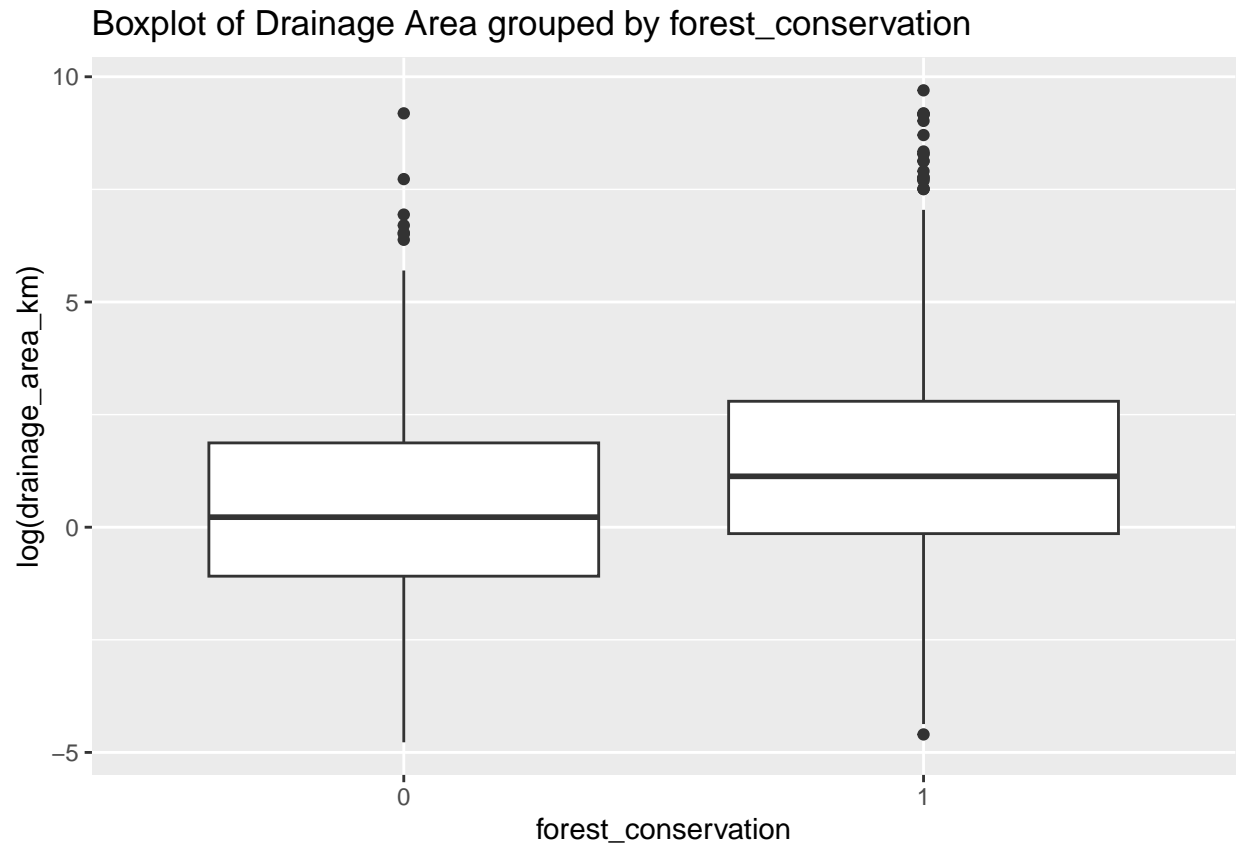
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(pasture_management)))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by pasture_management") +  
  xlab("pasture_management")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').



Effect of forest_conservation on drainage area, rainfall and vechval

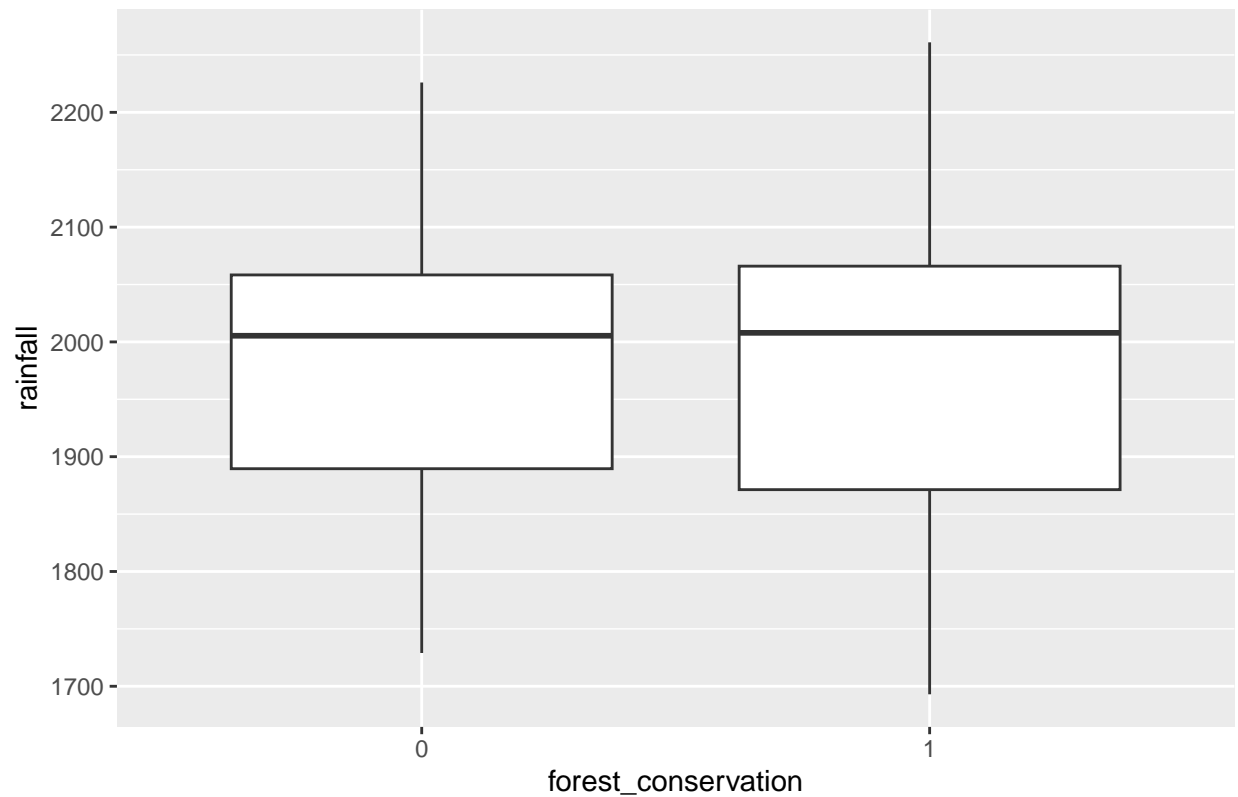
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(forest_conservation)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by forest_conservation") +
  xlab("forest_conservation")
```



```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x = as.factor(forest_conservation)))  
g + geom_boxplot() + labs(title = "Boxplot of rainfall grouped by forest_conservation") +  
  xlab("forest_conservation")
```

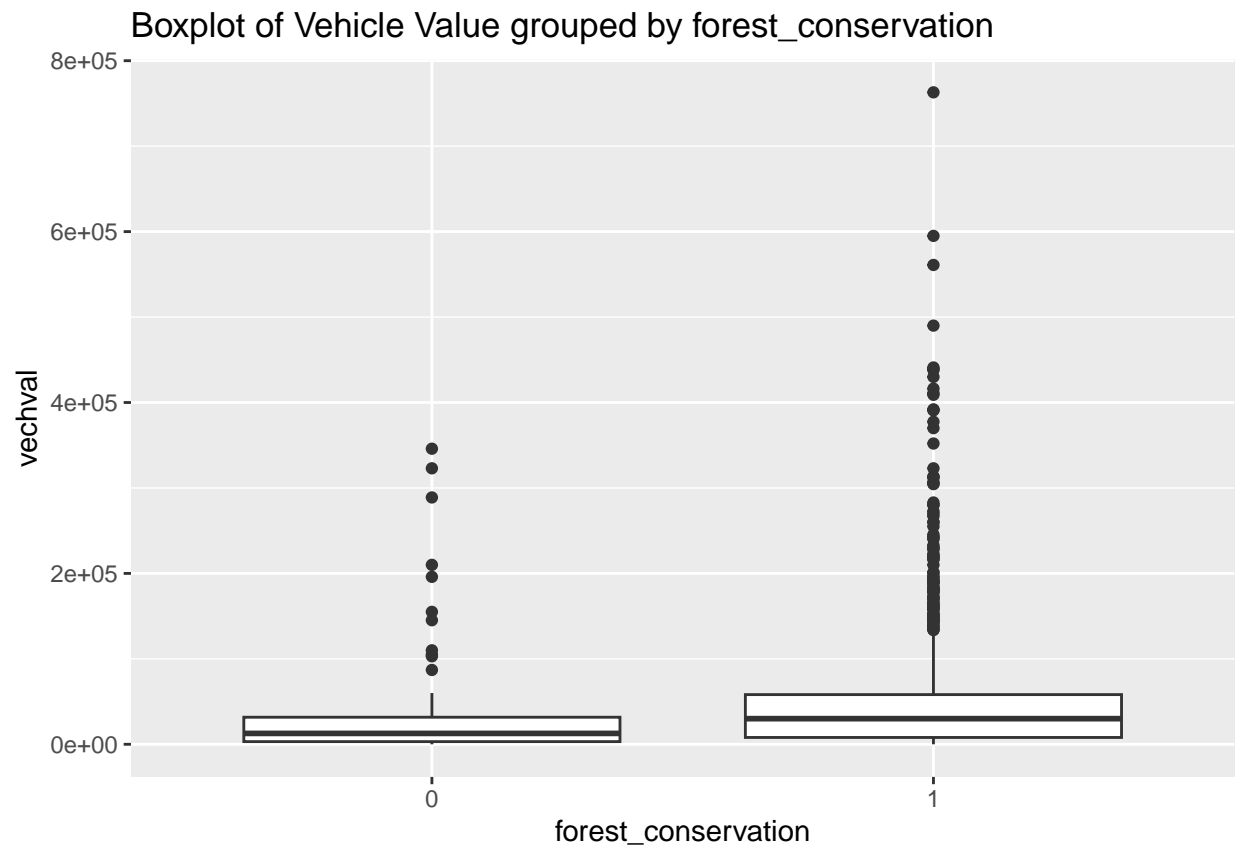
Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

Boxplot of rainfall grouped by forest_conservation



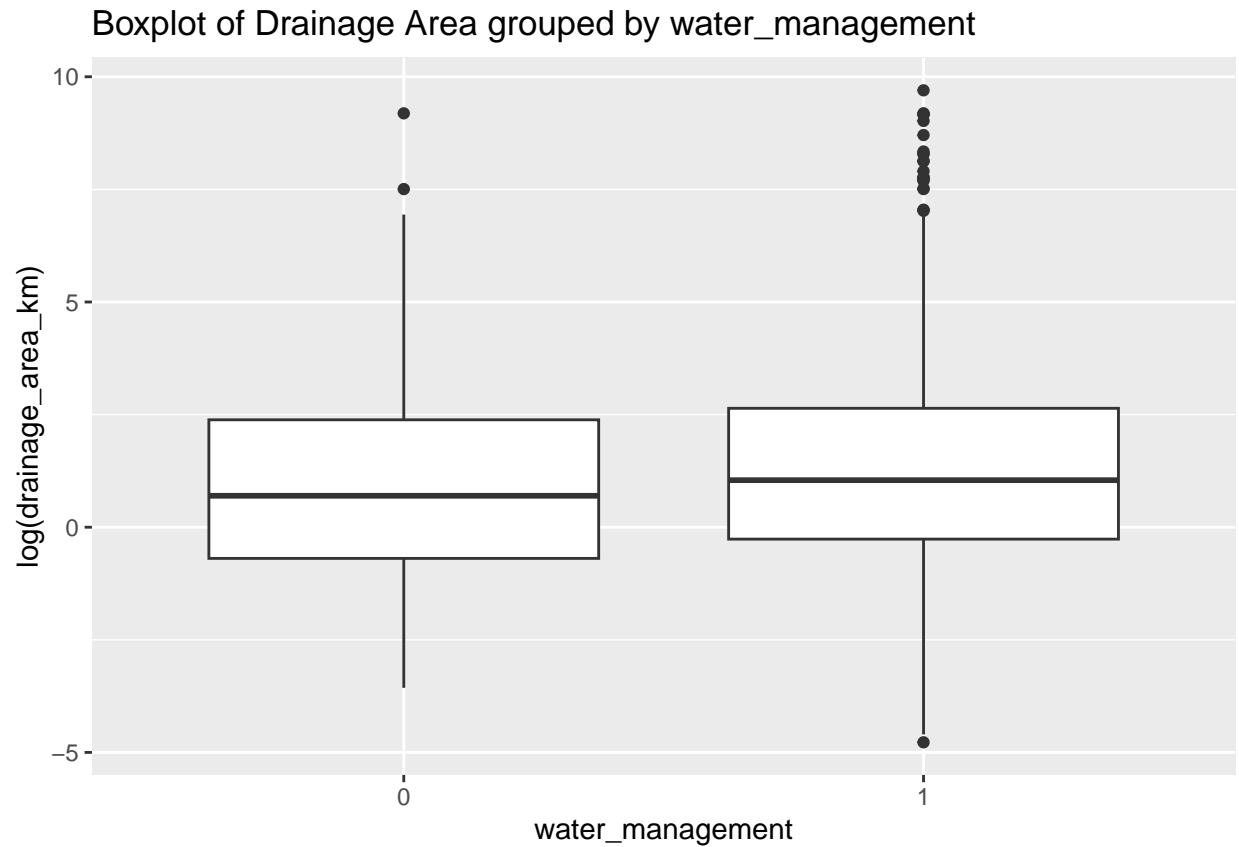
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(forest_conservation)))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by forest_conservation") +  
  xlab("forest_conservation")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').



Effect of water_management on drainage area, rainfall and vechval

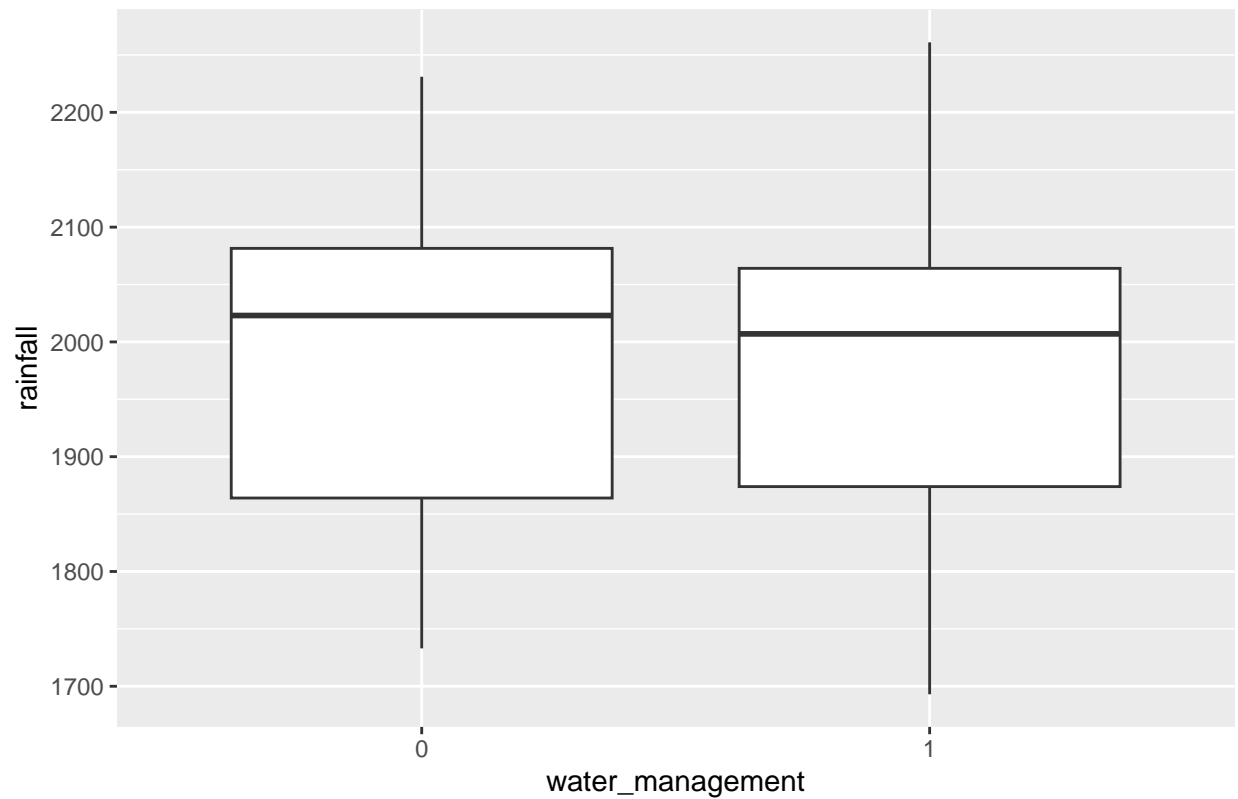
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(water_management)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by water_management") +
  xlab("water_management")
```



```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x = as.factor(water_management)))  
g + geom_boxplot() + labs(title = "Boxplot of rainfall grouped by water_management") +  
  xlab("water_management")
```

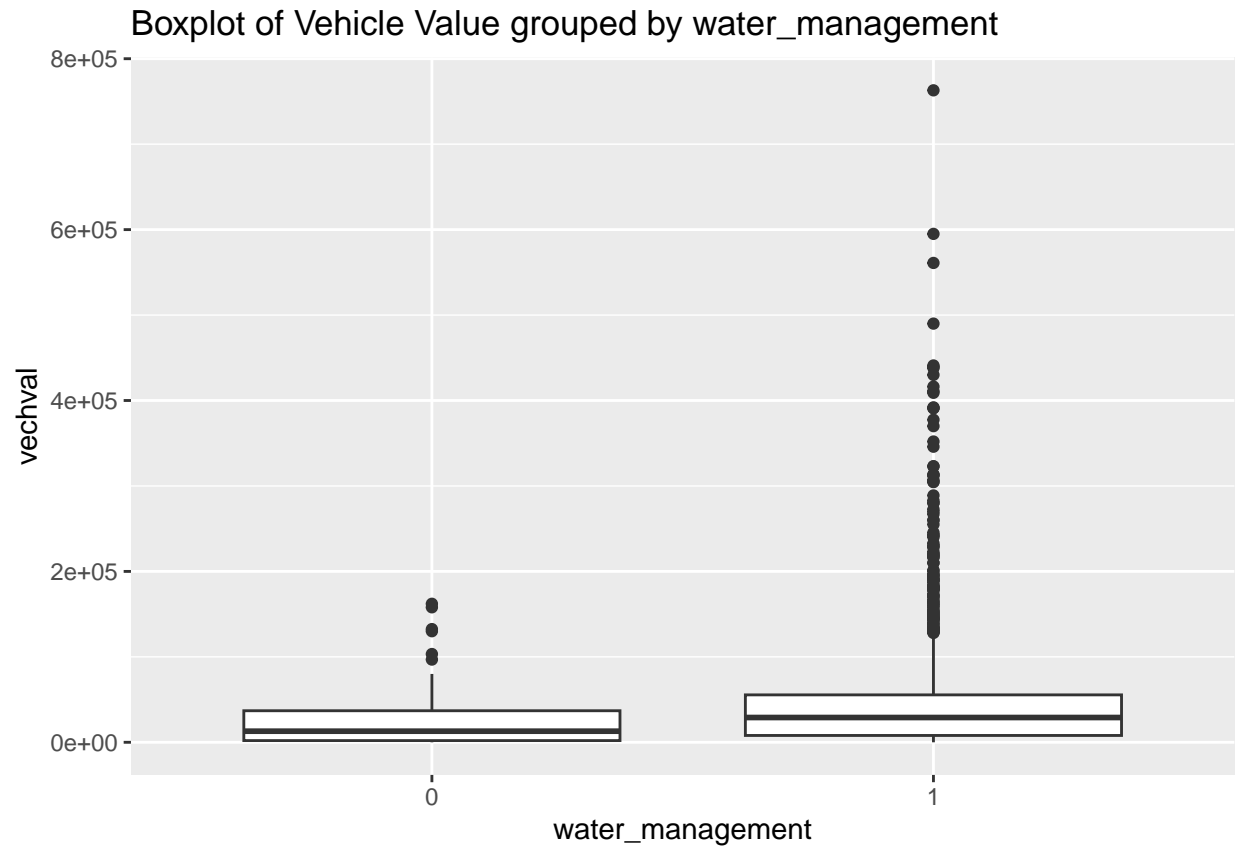
Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

Boxplot of rainfall grouped by water_management



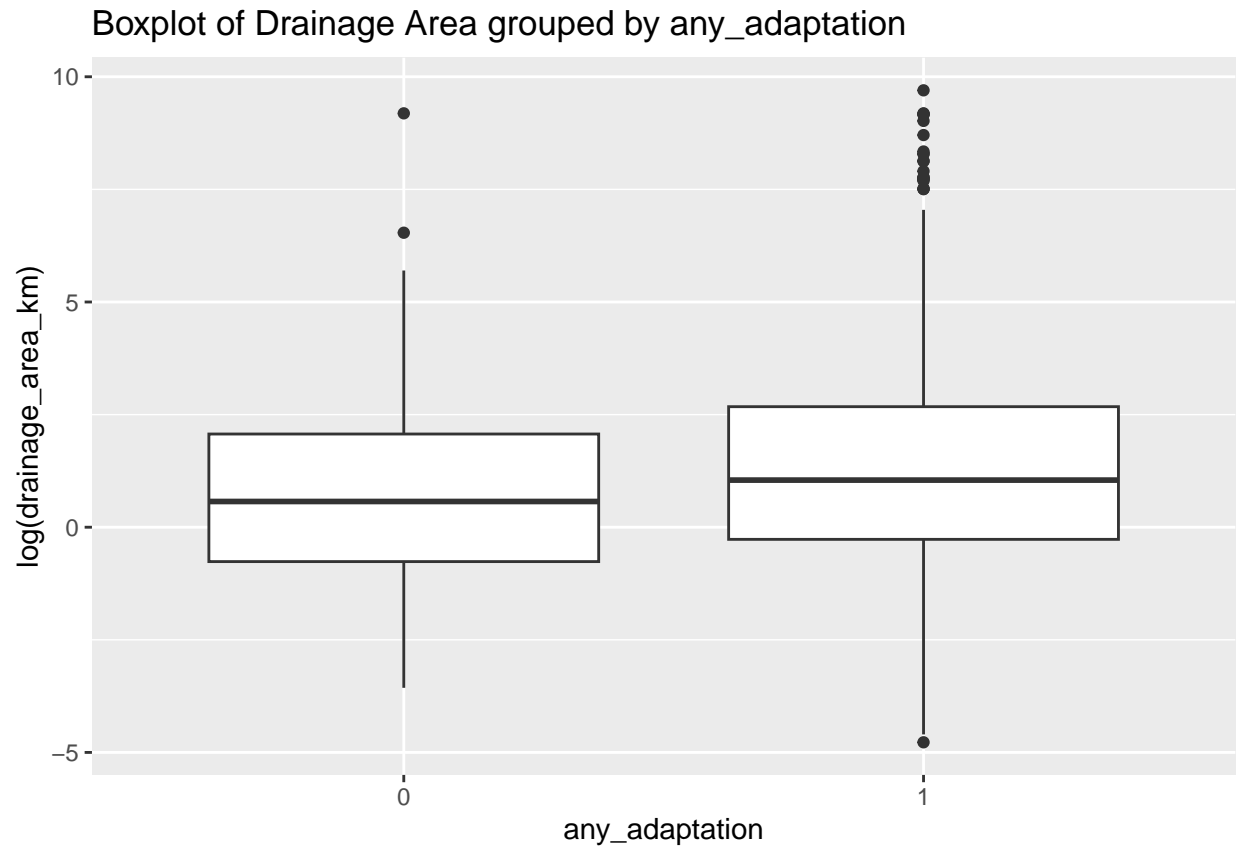
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(water_management)))  
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by water_management") +  
  xlab("water_management")
```

```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```

Effect of any_adaptation on drainage area, rainfall and vechval

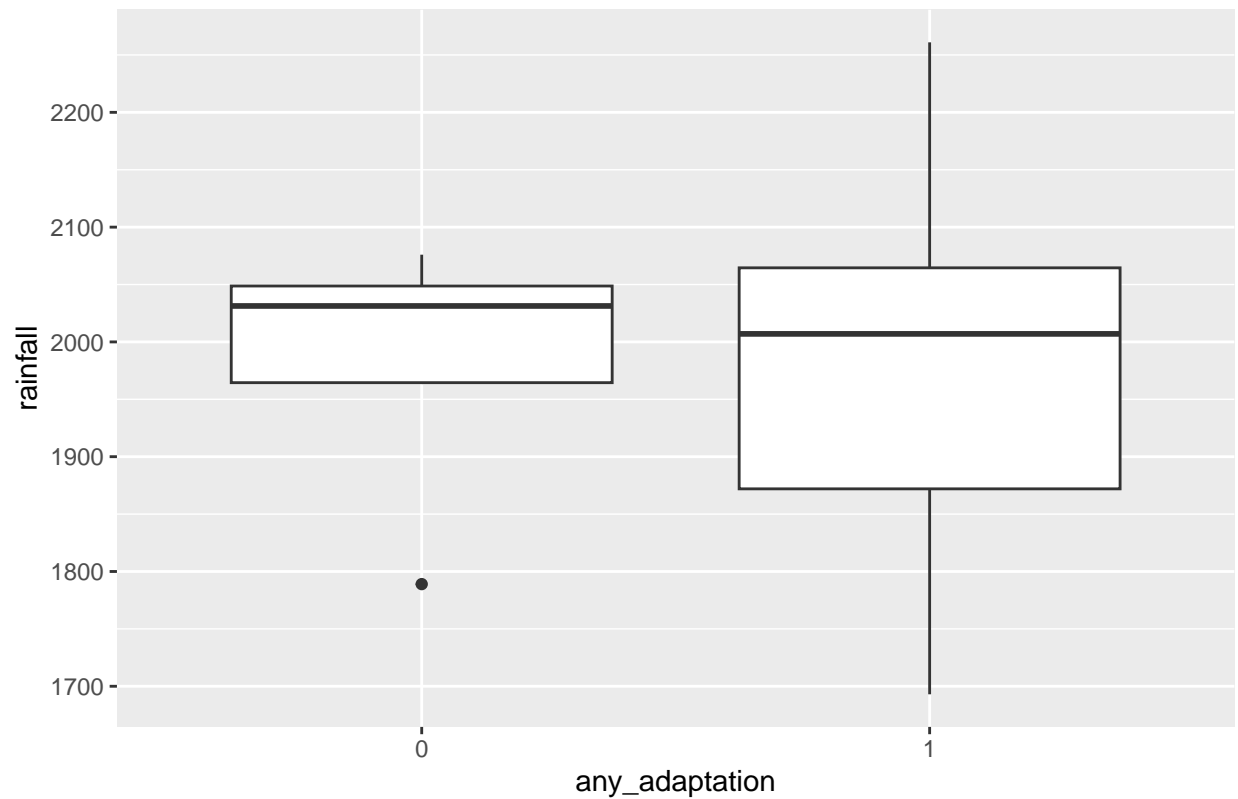
```
g <- ggplot(data = survey_final_gis, aes(y = log(drainage_area_km), x = as.factor(any_adaptation)))
g + geom_boxplot() + labs(title = "Boxplot of Drainage Area grouped by any_adaptation") +
  xlab("any_adaptation")
```



```
g <- ggplot(data = survey_final_gis, aes(y = rainfall, x = as.factor(any_adaptation)))  
g + geom_boxplot() + labs(title = "Boxplot of rainfall grouped by any_adaptation") +  
  xlab("any_adaptation")
```

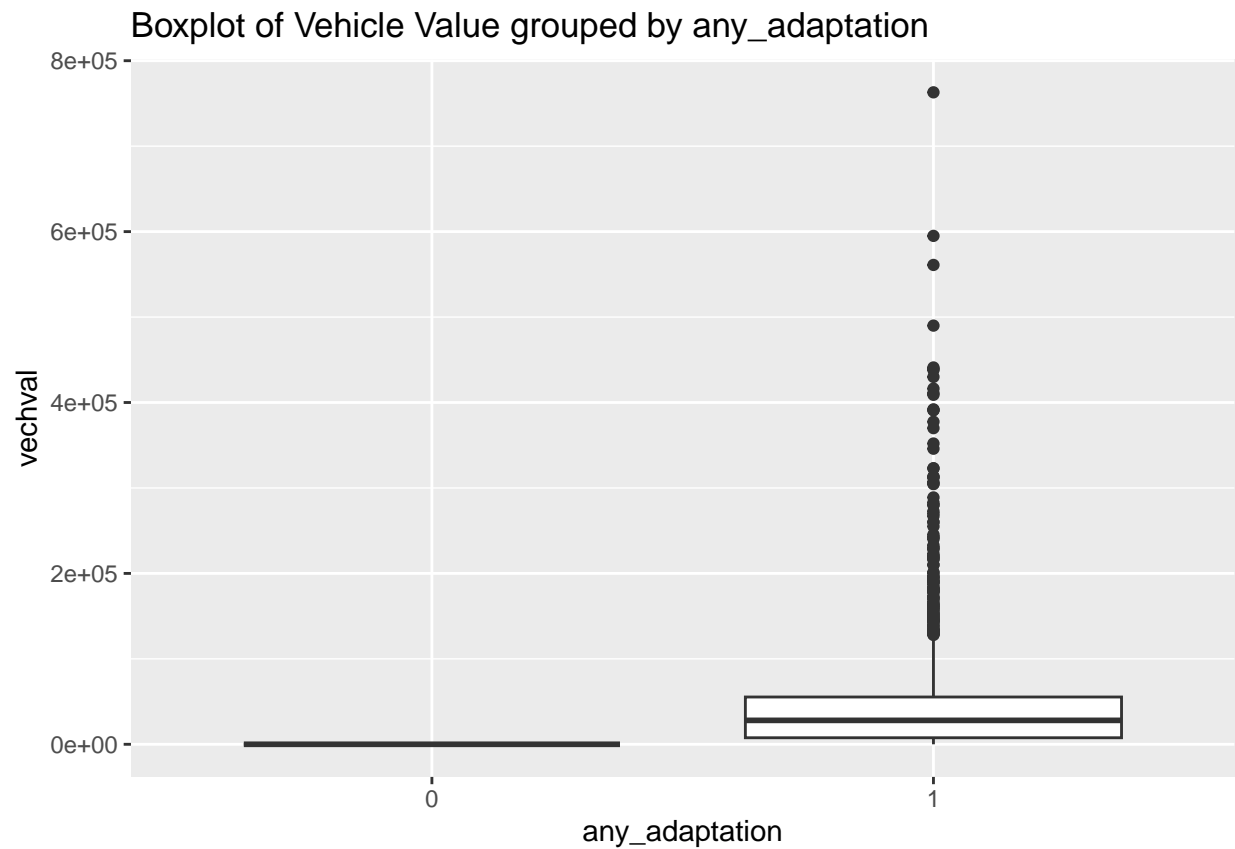
```
## Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').
```

Boxplot of rainfall grouped by any_adaptation



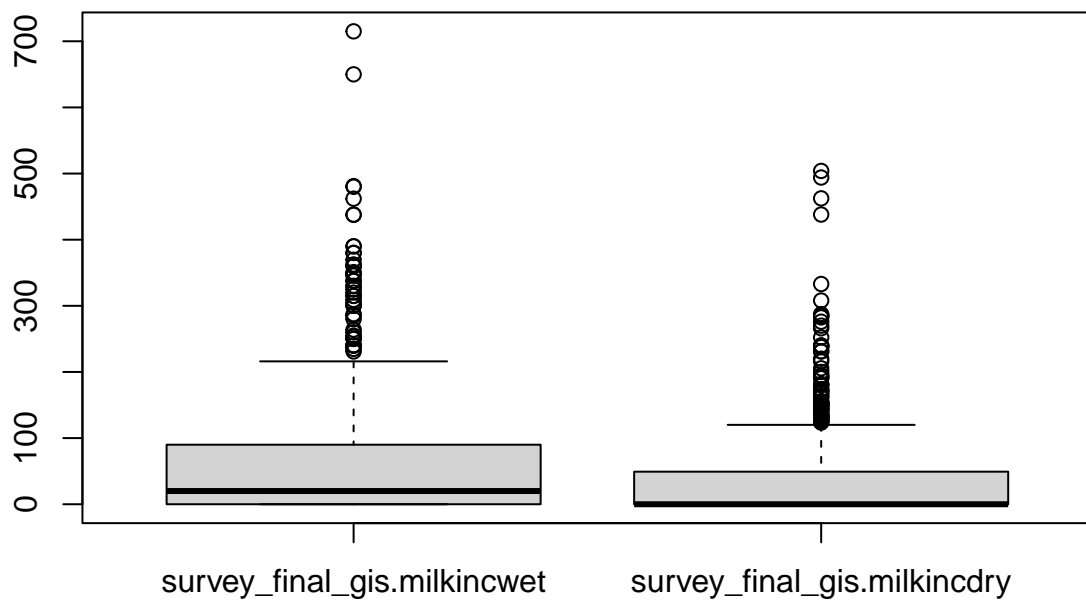
```
g <- ggplot(data = survey_final_gis, aes(y = vechval, x = as.factor(any_adaptation)))
g + geom_boxplot() + labs(title = "Boxplot of Vehicle Value grouped by any_adaptation") +
  xlab("any_adaptation")
```

Warning: Removed 140 rows containing non-finite values ('stat_boxplot()').

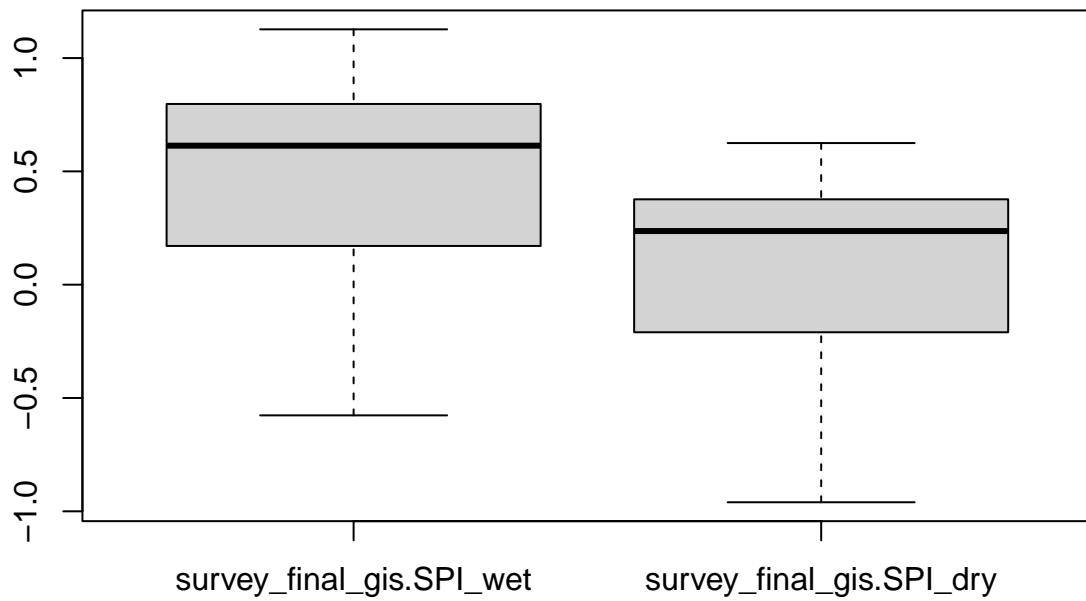


Boxplotx of milkincome, SPI and rainfall

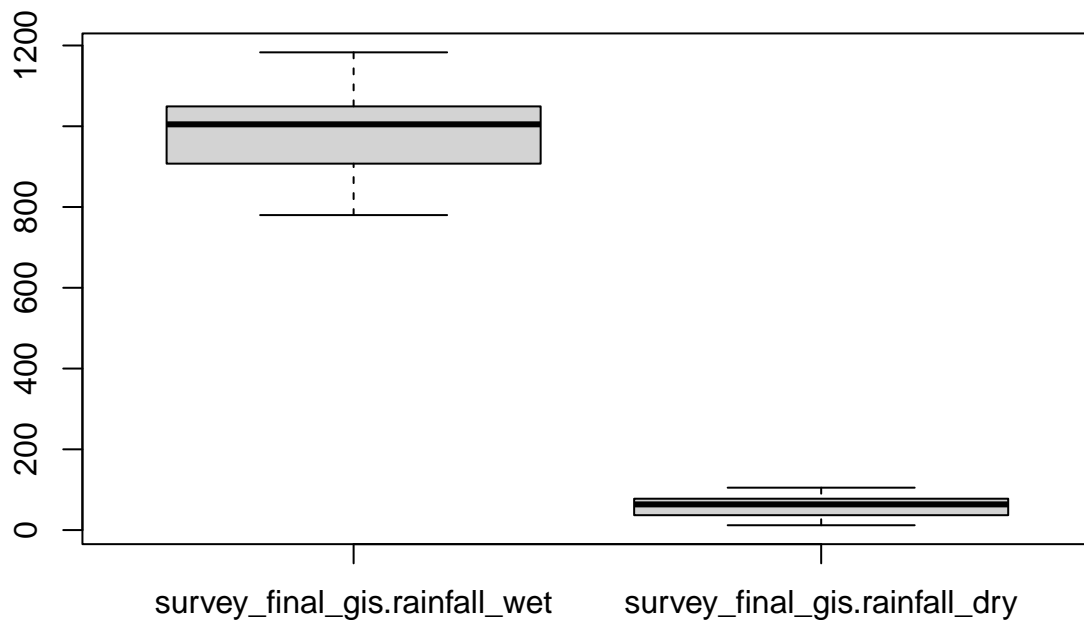
```
milkcinc <- data.frame(survey_final_gis$milkinwet,survey_final_gis$milkindry)
boxplot(milkcinc)
```



```
SPI <- data.frame(survey_final_gis$SPI_wet,survey_final_gis$SPI_dry)
boxplot(SPI)
```



```
rainfall <- data.frame(survey_final_gis$rainfall_wet,survey_final_gis$rainfall_dry)
boxplot(rainfall)
```



Statistcal analysis ## t test for drainage area

```

quartiles_1 <- quantile(survey_final_gis[survey_final_gis$any_adaptation==1,]$drainage_area_km, probs=c(0.25, 0.75))
IQR_1 <- IQR(survey_final_gis[survey_final_gis$any_adaptation==1,]$drainage_area_km, na.rm = TRUE)

Lower_1 <- quartiles_1[1] - 1.5*IQR_1
Upper_1 <- quartiles_1[2] + 1.5*IQR_1

data_no_outlier_1 <- subset(survey_final_gis[survey_final_gis$any_adaptation==1,]$drainage_area_km, survey_final_gis[survey_final_gis$any_adaptation==1,]$drainage_area_km > Upper_1 | survey_final_gis[survey_final_gis$any_adaptation==1,]$drainage_area_km < Lower_1)

quartiles_0 <- quantile(survey_final_gis[survey_final_gis$any_adaptation==0,]$drainage_area_km, probs=c(0.25, 0.75))
IQR_0 <- IQR(survey_final_gis[survey_final_gis$any_adaptation==0,]$drainage_area_km, na.rm = TRUE)

Lower_0 <- quartiles_0[1] - 1.5*IQR_0
Upper_0 <- quartiles_0[2] + 1.5*IQR_0

data_no_outlier_0 <- subset(survey_final_gis[survey_final_gis$any_adaptation==0,]$drainage_area_km, survey_final_gis[survey_final_gis$any_adaptation==0,]$drainage_area_km > Upper_0 | survey_final_gis[survey_final_gis$any_adaptation==0,]$drainage_area_km < Lower_0)

t.test(data_no_outlier_1, data_no_outlier_0, na.rm=TRUE, var.equal = FALSE)

```

```

##
## Welch Two Sample t-test
##
## data: data_no_outlier_1 and data_no_outlier_0
## t = 6.5409, df = 290.15, p-value = 2.756e-10
## alternative hypothesis: true difference in means is not equal to 0

```

```
## 95 percent confidence interval:
## 1.751338 3.258946
## sample estimates:
## mean of x mean of y
## 5.080324 2.575182
```

t test for rainfall

```
quartiles_1 <- quantile(survey_final_gis[survey_final_gis$any_adaptation==1,]$rainfall, probs=c(.25, .75))
IQR_1 <- IQR(survey_final_gis[survey_final_gis$any_adaptation==1,]$rainfall, na.rm = TRUE)

Lower_1 <- quartiles_1[1] - 1.5*IQR_1
Upper_1 <- quartiles_1[2] + 1.5*IQR_1

data_no_outlier_1 <- subset(survey_final_gis[survey_final_gis$any_adaptation==1,]$rainfall, survey_final_gis$any_adaptation==1)

quartiles_0 <- quantile(survey_final_gis[survey_final_gis$any_adaptation==0,]$rainfall, probs=c(.25, .75))
IQR_0 <- IQR(survey_final_gis[survey_final_gis$any_adaptation==0,]$rainfall, na.rm = TRUE)

Lower_0 <- quartiles_0[1] - 1.5*IQR_0
Upper_0 <- quartiles_0[2] + 1.5*IQR_0

data_no_outlier_0 <- subset(survey_final_gis[survey_final_gis$any_adaptation==0,]$rainfall, survey_final_gis$any_adaptation==0)

t.test(data_no_outlier_1, data_no_outlier_0, na.rm=TRUE, var.equal = FALSE)
```

```
##
## Welch Two Sample t-test
##
## data: data_no_outlier_1 and data_no_outlier_0
## t = -3.9741, df = 2.1904, p-value = 0.04973
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -127.0720236 -0.1718252
## sample estimates:
## mean of x mean of y
## 1982.589 2046.211
```

t test for milkincome at different seasons

We would like to see if there is any difference of milkincome between dry and wet season. The null hypothesis is mean value of milkincome is the same for dry and wet season. $H_0: \mu_{mlkinc,dry} = \mu_{mlkinc,wet}$ vs $H_a: \mu_{mlkinc,dry} \neq \mu_{mlkinc,wet}$

```
t.test(survey_final_gis_clean$milkincwet, survey_final_gis_clean$milkincdry, na.rm=TRUE, var.equal = FALSE)

##
## Welch Two Sample t-test
##
## data: survey_final_gis_clean$milkincwet and survey_final_gis_clean$milkincdry
```



```
## t = 7.5306, df = 1816.9, p-value = 7.911e-14
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  18.16078 30.95159
## sample estimates:
## mean of x mean of y
##  57.62763  33.07144
```

As p is much less than 0.05, we are confident the mean value of milkincome of wetseason is significant higher than dry season.

t-test for rainfall at different seasons

```
t.test(survey_final_gis_clean$rainfall_wet,survey_final_gis_clean$rainfall_dry,na.rm=TRUE,var.equal = F)

##
## Welch Two Sample t-test
##
## data: survey_final_gis_clean$rainfall_wet and survey_final_gis_clean$rainfall_dry
## t = 347.03, df = 1375.1, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  920.2497 930.7128
## sample estimates:
## mean of x mean of y
## 984.50655  59.02528
```

t-test for SPI at different seasons

We would like to see if there is any difference in SPI between dry and wet season. The null hypothesis is mean value of SPI is the same for dry and wet season. $H_0: \mu_{SPI,dry} = \mu_{SPI,wet}$ vs $H_a: \mu_{SPI,dry} \neq \mu_{SPI,wet}$

```
t.test(survey_final_gis_clean$SPI_wet,survey_final_gis_clean$SPI_dry,na.rm=TRUE,var.equal = FALSE)

##
## Welch Two Sample t-test
##
## data: survey_final_gis_clean$SPI_wet and survey_final_gis_clean$SPI_dry
## t = 22.235, df = 2328.1, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3328086 0.3971907
## sample estimates:
## mean of x mean of y
## 0.4610647 0.0960650
```

Average SPI and average monthly SPI in the peak of wet season is significantly higher than those of dry season.

Anova analysis

We hypothesize drainage area of three regions may be different. We propose null hypothesis is drainage area means of three regions are the same. The alternative hypothesis is drainage area means of three regions are different.

effect of region on drainage_area_km

```
survey_final_gis$studycode<- factor(survey_final_gis$region)
aov_drainage <- aov(drainage_area_km ~ region, data=survey_final_gis)

summary(aov_drainage)
```

```
##              Df      Sum Sq Mean Sq F value Pr(>F)
## region        2    4121158 2060579   2.871 0.0571 .
## Residuals    1197 859256274  717841
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 140 observations deleted due to missingness
```

```
TukeyHSD(aov_drainage, conf.level=.95)
```

```
##    Tukey multiple comparisons of means
##      95% family-wise confidence level
##
## Fit: aov(formula = drainage_area_km ~ region, data = survey_final_gis)
##
## $region
##              diff          lwr          upr      p adj
## Ouro Preto do Oeste-Ariquemes    -59.13236 -199.507127  81.24241 0.5842723
## Rolim de Moura-Ariquemes         81.66583  -62.733974 226.06563 0.3803196
## Rolim de Moura-Ouro Preto do Oeste 140.79819   2.692364 278.90401 0.0445052
```

As p-value of F test is extremely small, we are confident that the drainage means of three regions are different. We further compared drainage means of three regions and found region “Rolim de Moura” has the highest drainage area, OPO has the smallest.

effect of region on income from beef

We then wants to see if there is difference in income from beef at different regions.

```
aov_incbeef <- aov(incbeef ~ region, data=survey_final_gis )
summary(aov_incbeef)
```

```
##              Df      Sum Sq  Mean Sq F value Pr(>F)
## region        2 1.371e+10 6.853e+09   1.124  0.325
## Residuals    695 4.236e+12 6.096e+09
## 642 observations deleted due to missingness
```

```
TukeyHSD(aov_incbeef, conf.level=.95)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = incbeef ~ region, data = survey_final_gis)
##
## $region
##              diff          lwr          upr      p adj
## Ouro Preto do Oeste-Ariquemes -11239.618 -29238.61  6759.375 0.3076587
## Rolim de Moura-Ariquemes      -6577.226 -22674.83  9520.381 0.6026638
## Rolim de Moura-Ouro Preto do Oeste  4662.393 -13029.43 22354.211 0.8097785
```

As p value is larger than 0.05, we can not reject our hypothesis that the means of income from beef at different regions are the same.

effect of region on rainfall

```
aov_rainfall <- aov(rainfall ~ region, data=survey_final_gis )
summary(aov_rainfall)
```

```
##              Df   Sum Sq Mean Sq F value Pr(>F)
## region         2 12088742 6044371    1703 <2e-16 ***
## Residuals    1197  4249056    3550
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 140 observations deleted due to missingness
```

```
TukeyHSD(aov_rainfall, conf.level=.95)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = rainfall ~ region, data = survey_final_gis)
##
## $region
##              diff          lwr          upr      p adj
## Ouro Preto do Oeste-Ariquemes    7.163234   -2.708057   17.03452 0.2045033
## Rolim de Moura-Ariquemes     -210.154234  -220.308569 -199.99990 0.0000000
## Rolim de Moura-Ouro Preto do Oeste -217.317468 -227.029204 -207.60573 0.0000000
```

As p-value for F test of AONVA is very small, rainfall in three regions is not the same. It can be seen the rainfall in region “Rolim de Moura” is less than that in the other two.

effect of region on wealth

```
aov_vechval <- aov(vechval ~ region, data=survey_final_gis_clean)
summary(aov_vechval)
```

```
##              Df      Sum Sq   Mean Sq F value    Pr(>F)
## region         2 1.581e+11  7.907e+10   14.43 6.41e-07 ***
## Residuals    1197 6.558e+12  5.479e+09
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(aov_vechval, conf.level=.95)
```

```
##      Tukey multiple comparisons of means
##      95% family-wise confidence level
##
## Fit: aov(formula = vechval ~ region, data = survey_final_gis_clean)
##
## $region
##              diff              lwr              upr              p adj
## Ouro Preto do Oeste-Ariquemes -15301.14 -27564.9489 -3037.328 0.0097323
## Rolim de Moura-Ariquemes      12195.46  -420.0017 24810.912 0.0606958
## Rolim de Moura-Ouro Preto do Oeste 27496.59 15431.0086 39562.178 0.0000003
```

The family wealth is significant different in three regions. The wealth in region “3” Rolim de Moura” is the largest.

Effect of region on adaptation method

Will need to do this for each method, since methods can overlap

First - cattle management

```
# Need to change studycode into a factor
survey_final_gis$studycode <- as.factor(survey_final_gis$studycode)
aov_cattle_management <- aov(cattle_management ~ studycode, data = survey_final_gis)
summary(aov_cattle_management)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## studycode      2   0.99  0.4958   2.042   0.13
## Residuals    1197 290.68  0.2428
## 140 observations deleted due to missingness
```

Cattle management appears to be the same across all three regions based on the p-value

Next - water management

```
aov_water_management <- aov(water_management ~ studycode, data = survey_final_gis)
summary(aov_water_management)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## studycode      2   0.55  0.27330   6.775 0.00119 **
```

```
## Residuals    1197    48.29 0.04034
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 140 observations deleted due to missingness
```

```
TukeyHSD(aov_water_management, conf.level=.95)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = water_management ~ studycode, data = survey_final_gis)
##
## $studycode
##
##           diff           lwr           upr
## Ouro Preto do Oeste-Ariquemes -0.05217268 -0.08544920 -0.018896162
## Rolim de Moura-Ariquemes      -0.02989130 -0.06412198  0.004339368
## Rolim de Moura-Ouro Preto do Oeste 0.02228138 -0.01045728  0.055020031
##
##           p adj
## Ouro Preto do Oeste-Ariquemes 0.0007144
## Rolim de Moura-Ariquemes      0.1011412
## Rolim de Moura-Ouro Preto do Oeste 0.2472685
```

Based on the p-value, we know at least two regions are significantly different Using the results of Tukey's HSD, we can see that regions 1 and 2 (Ariquemes and OPO) are significantly different when it comes to water management

Next - pasture management

```
aov_pasture_management <- aov(pasture_management ~ studycode, data = survey_final_gis)
summary(aov_pasture_management)
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## studycode      2   3.81   1.9064    8.327 0.000256 ***
## Residuals  1197 274.05   0.2289
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 140 observations deleted due to missingness
```

```
TukeyHSD(aov_pasture_management, conf.level=.95)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = pasture_management ~ studycode, data = survey_final_gis)
##
## $studycode
##
##           diff           lwr           upr
## Ouro Preto do Oeste-Ariquemes -0.08256310 -0.16183872 -0.003287472
## Rolim de Moura-Ariquemes       0.05131074 -0.03023799  0.132859475
## Rolim de Moura-Ouro Preto do Oeste 0.13387384  0.05587959  0.211868094
##
##           p adj
## Ouro Preto do Oeste-Ariquemes 0.0389323
## Rolim de Moura-Ariquemes      0.3025965
## Rolim de Moura-Ouro Preto do Oeste 0.0001767
```

From the result of the ANOVA, we can see there is a difference between at least one region. Looking at the result of Tukey, we can see the difference is significant between regions 1 and 2, and regions 2 and 3

Next - forest conservation

```
aov_forest_conservation <- aov(forest_conservation ~ studycode, data = survey_final_gis)
summary(aov_forest_conservation)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## studycode      2   0.33   0.1644   1.976  0.139
## Residuals    1197  99.59   0.0832
## 140 observations deleted due to missingness
```

```
#TukeyHSD(aov_pasture_management, conf.level=.95)
```

There is not a significant difference amongst regions due to forest conservation

Effect of wealth on number of adaptations

```
survey_final_gis <- survey_final_gis %>% mutate(adaptation_count = cattle_management + pasture_management)
aov_vechval_adaptation_count <- aov(vechval ~ adaptation_count, data = survey_final_gis)
summary(aov_vechval_adaptation_count)
```

```
##              Df      Sum Sq   Mean Sq F value Pr(>F)
## adaptation_count      1 4.354e+11 4.354e+11   83.05 <2e-16 ***
## Residuals           1198 6.281e+12 5.243e+09
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 140 observations deleted due to missingness
```

```
fit_lr <- lm(adaptation_count ~ vechval, data = survey_final_gis)
summary(fit_lr)
```

```
##
## Call:
## lm(formula = adaptation_count ~ vechval, data = survey_final_gis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.7671 -0.7821  0.1191  0.8003  1.2329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.767e+00  2.965e-02  93.340   <2e-16 ***
## vechval      2.993e-06  3.284e-07   9.113   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8511 on 1198 degrees of freedom
## (140 observations deleted due to missingness)
## Multiple R-squared:  0.06483,    Adjusted R-squared:  0.06405
## F-statistic: 83.05 on 1 and 1198 DF,  p-value: < 2.2e-16
```

Modelling

Multiple Logistic Regression Model

First to create a few different MLRs to see what the relationship is between adaptation methods and water availability.

Model 1

In this model, I will select the largest amount of variables. The dependent variable will be the general adaptation variable. The independent variables will be: - Drainage area - Lot size (can't find in data - ask Mariana) - Soil type (100% missing - so maybe not) - Vehicle value - SPI - using the maximum in the year - Risk - Lot value

(This model did not converge)

Model 2

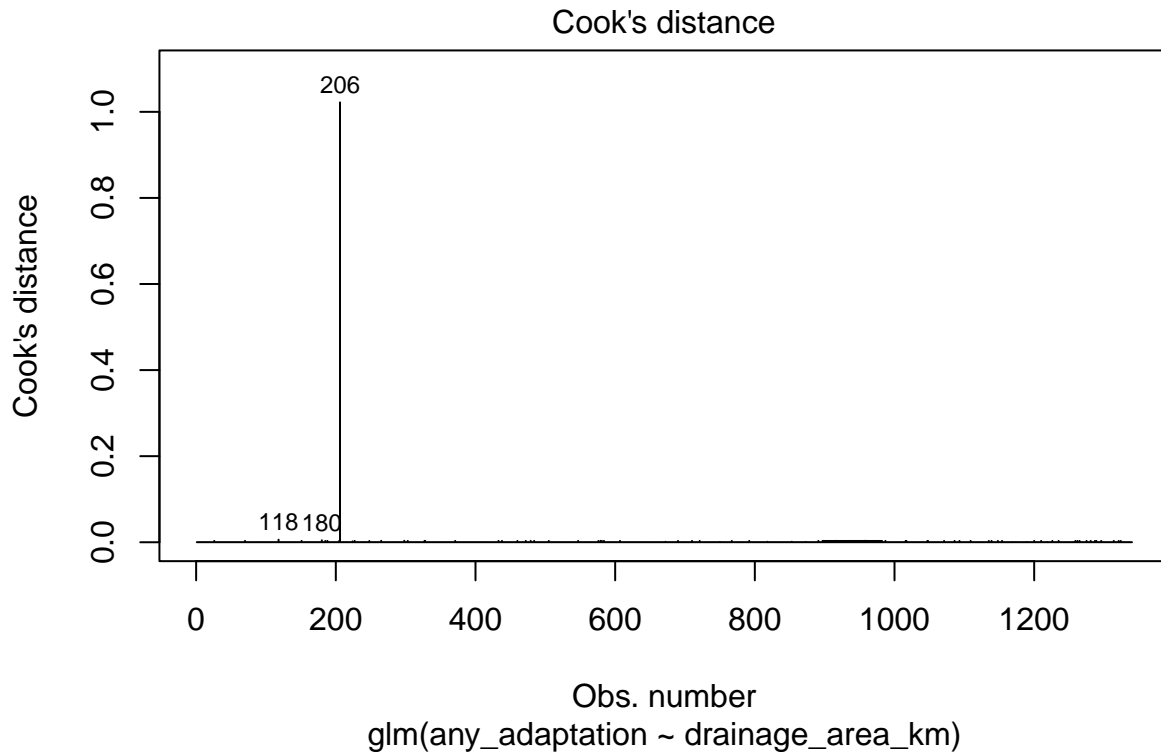
Trying with only drainage area to see if there's a relationship since previous model did not converge

```
fit_2 <- glm(any_adaptation ~ drainage_area_km, family = binomial, data = survey_final_gis)
summary(fit_2)
```

```
##
## Call:
## glm(formula = any_adaptation ~ drainage_area_km, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.1088684   0.0891751   23.65  <2e-16 ***
## drainage_area_km 0.0000746   0.0001408    0.53   0.596
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 914.36  on 1339  degrees of freedom
## Residual deviance: 914.01  on 1338  degrees of freedom
## AIC: 918.01
##
## Number of Fisher Scoring iterations: 5
```

The drainage area is not a statistically significant parameter in this case. However, there are outliers in the data that might be affecting this. Let's investigate.

```
plot(fit_2, which = 4)
```



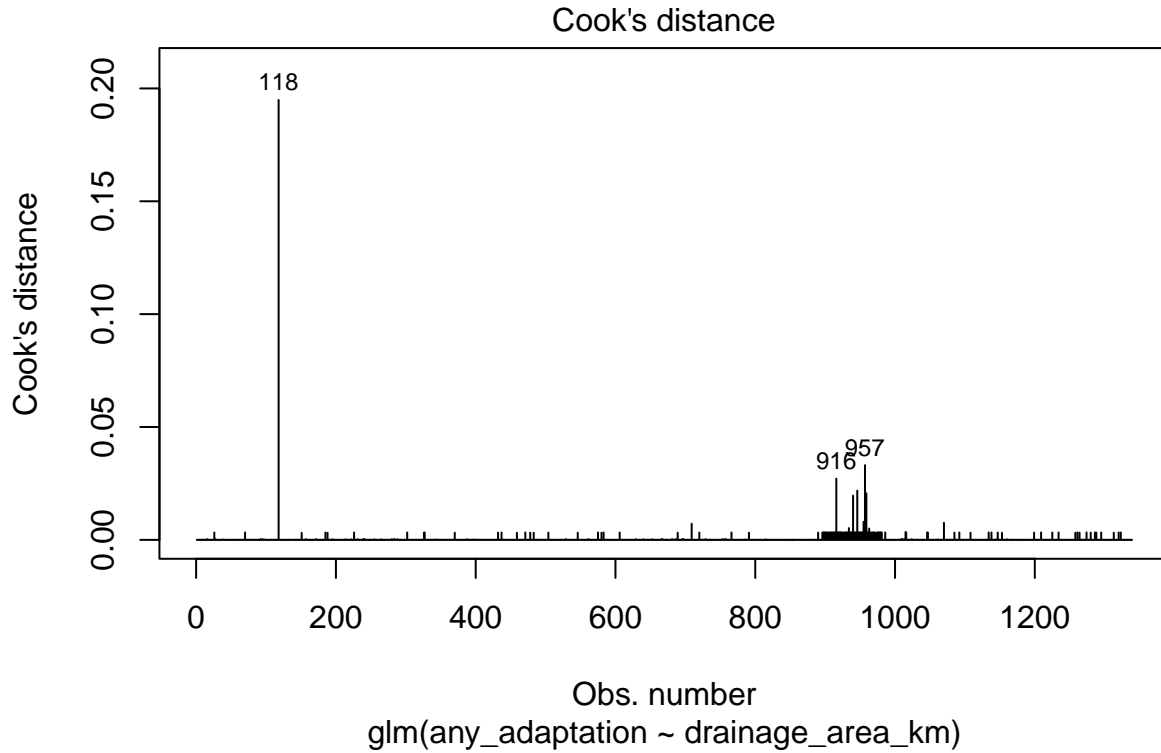
Fitting the model without observation 206

```
survey_final_gis_no <- survey_final_gis[-c(206),]
fit_3 <- glm(any_adaptation ~ drainage_area_km, family = binomial, data = survey_final_gis_no)
summary(fit_3)
```

```
##
## Call:
## glm(formula = any_adaptation ~ drainage_area_km, family = binomial,
##      data = survey_final_gis_no)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.0599143  0.0913975  22.538  <2e-16 ***
## drainage_area_km 0.0015679  0.0009512   1.648  0.0993 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 909.89  on 1338  degrees of freedom
## Residual deviance: 902.26  on 1337  degrees of freedom
## AIC: 906.26
##
## Number of Fisher Scoring iterations: 8
```



```
plot(fit_3, which = 4)
```

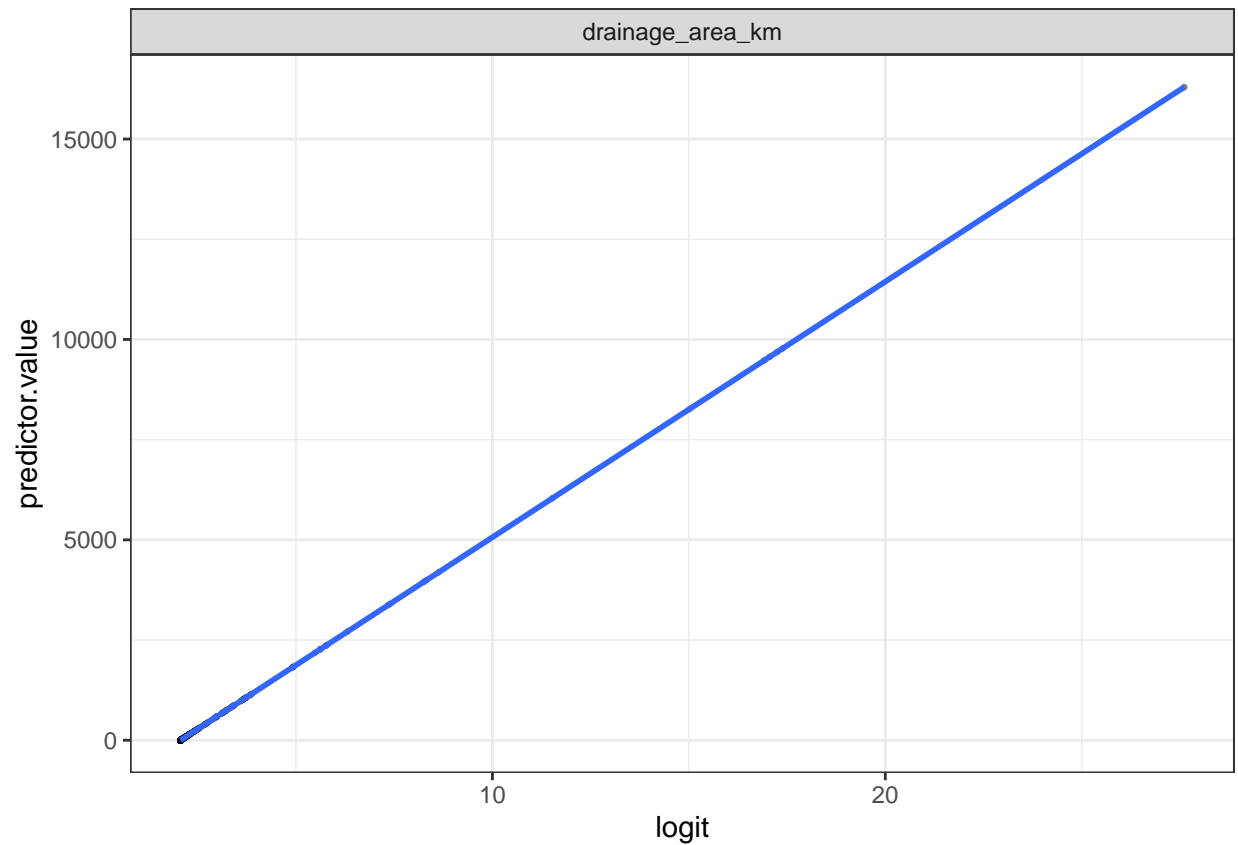


Now to check Log Reg assumptions

```
probabilities <- predict(fit_3, type = "response")
# Select only numeric predictors
mydata <- survey_final_gis_no %>%
  dplyr::select_if(is.numeric) %>%
  select(drainage_area_km)
predictors <- colnames(mydata)
# Bind the logit and tidying the data for plot
mydata <- mydata %>%
  mutate(logit = log(probabilities/(1-probabilities))) %>%
  gather(key = "predictors", value = "predictor.value", -logit)
```

```
ggplot(mydata, aes(logit, predictor.value)) +
  geom_point(size = 0.5, alpha = 0.5) +
  geom_smooth(method = "loess") +
  theme_bw() +
  facet_wrap(~predictors, scales = "free_y")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



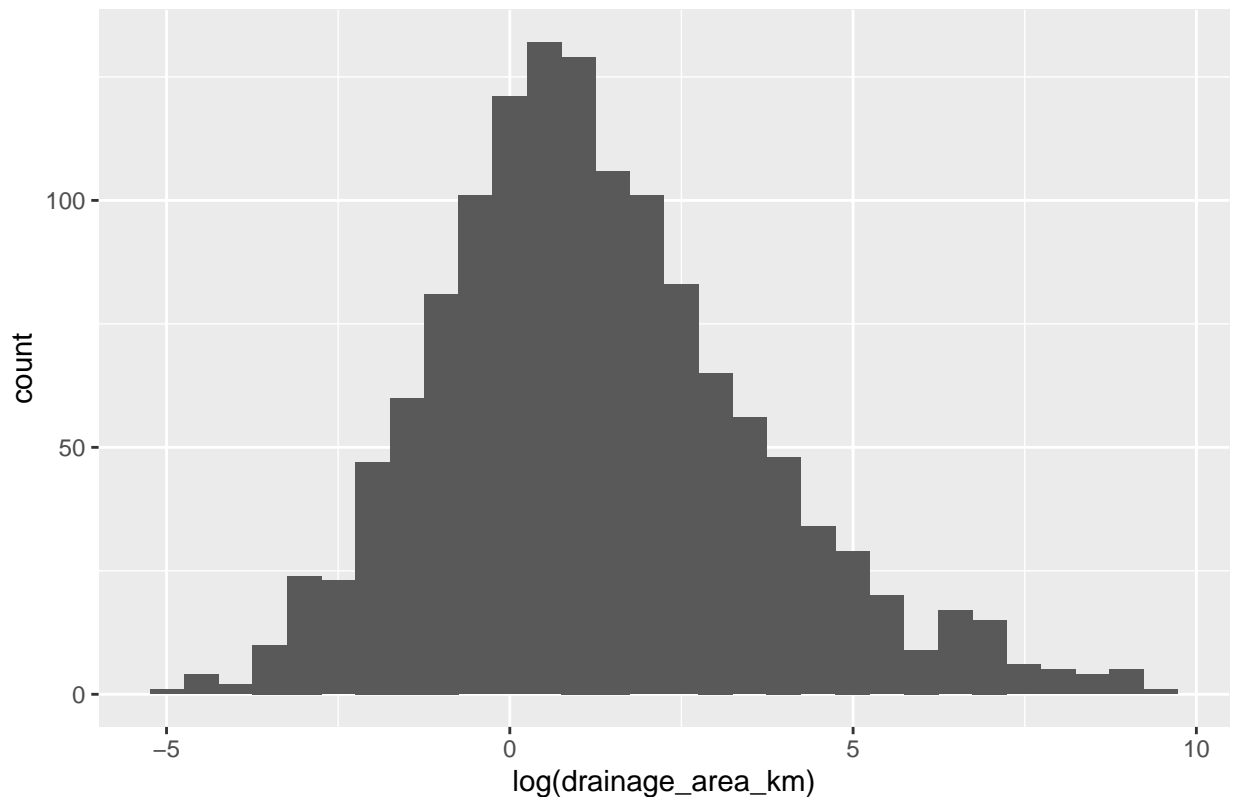
Model 3 - Transforming Drainage Area

Let's look at the distribution of drainage area to see if we should transform it

```
g <- ggplot(data = survey_final_gis_no, aes(x = log(drainage_area_km)))  
g + geom_histogram() + labs(title = "Histogram of log(Drainage Area)")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

Histogram of log(Drainage Area)



Now to create the model:

```
fit_transform <- glm(any_adaptation ~ log(drainage_area_km), family = binomial, data = survey_final_gis)
summary(fit_transform)
```

```
##
## Call:
## glm(formula = any_adaptation ~ log(drainage_area_km), family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      2.00718    0.09431  21.282 < 2e-16 ***
## log(drainage_area_km) 0.10519    0.03962   2.655  0.00792 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 914.36  on 1339  degrees of freedom
## Residual deviance: 906.98  on 1338  degrees of freedom
## AIC: 910.98
##
## Number of Fisher Scoring iterations: 5
```

Model 4

For this model, we will take Dr. Harris's suggestion of an indicator variable for the large drainage area lots (I believe is what he meant)

First going to see the fit with outliers chopped off

```
survey_final_gis_filtered <- survey_final_gis %>% filter(drainage_area_km < 500)
fit_4 <- glm(any_adaptation ~ drainage_area_km, data = survey_final_gis_filtered, family = binomial)
summary(fit_4)
```

```
##
## Call:
## glm(formula = any_adaptation ~ drainage_area_km, family = binomial,
##      data = survey_final_gis_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.0771756   0.0949895  21.867   <2e-16 ***
## drainage_area_km 0.0004578   0.0018537   0.247    0.805
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 893.26  on 1284  degrees of freedom
## Residual deviance: 893.20  on 1283  degrees of freedom
## AIC: 897.2
##
## Number of Fisher Scoring iterations: 4
```

model 5

```
survey_final_gis2<- drop_na(survey_final_gis,SPI_year)
survey_final_gis2$rainfall2 <- scale(survey_final_gis2$rainfall,center=TRUE,scale=T)
survey_final_gis2$drainage_area_km2 <- scale(survey_final_gis2$drainage_area_km,center=TRUE,scale=T)
fit_3 <- glm(any_adaptation ~ log(drainage_area_km) + log(rainfall), family = binomial,data = survey_f
summary(fit_3)
```

```
##
## Call:
## glm(formula = any_adaptation ~ log(drainage_area_km) + log(rainfall),
##      family = binomial, data = survey_final_gis2)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)         9.5080    63.5680   0.150   0.881
## log(drainage_area_km) -0.1858     0.1883  -0.987   0.324
## log(rainfall)        -0.4552     8.3677  -0.054   0.957
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
## Null deviance: 53.617 on 1199 degrees of freedom
## Residual deviance: 52.706 on 1197 degrees of freedom
## AIC: 58.706
##
## Number of Fisher Scoring iterations: 8
```

Now look at individual adaptation variables

```
fit_cattle <- glm(cattle_management ~ drainage_area_km, family = binomial, data = survey_final_gis)
summary(fit_cattle)
```

```
##
## Call:
## glm(formula = cattle_management ~ drainage_area_km, family = binomial,
## data = survey_final_gis)
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.298e-01 5.717e-02 -9.267 <2e-16 ***
## drainage_area_km 8.492e-05 6.657e-05 1.276 0.202
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1770.4 on 1339 degrees of freedom
## Residual deviance: 1768.7 on 1338 degrees of freedom
## AIC: 1772.7
##
## Number of Fisher Scoring iterations: 4
```

```
fit_cattle_log = glm(cattle_management ~ log(drainage_area_km), family = binomial, data = survey_final_gis)
summary(fit_cattle_log)
```

```
##
## Call:
## glm(formula = cattle_management ~ log(drainage_area_km), family = binomial,
## data = survey_final_gis)
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.58609 0.06488 -9.033 <2e-16 ***
## log(drainage_area_km) 0.05162 0.02374 2.174 0.0297 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1770.4 on 1339 degrees of freedom
## Residual deviance: 1765.7 on 1338 degrees of freedom
## AIC: 1769.7
##
## Number of Fisher Scoring iterations: 4
```

Saw significance when using log(drainage area)

Next lets look at pasture management

```
fit_pasture <- glm(pasture_management ~ drainage_area_km, family = binomial, data = survey_final_gis)
summary(fit_pasture)
```

```
##
## Call:
## glm(formula = pasture_management ~ drainage_area_km, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.674e-01  5.583e-02   4.790 1.66e-06 ***
## drainage_area_km 1.028e-04  7.926e-05   1.296   0.195
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1831.7  on 1339  degrees of freedom
## Residual deviance: 1829.7  on 1338  degrees of freedom
## AIC: 1833.7
##
## Number of Fisher Scoring iterations: 4
```

```
fit_pasture_log <- glm(pasture_management ~ log(drainage_area_km), family = binomial, data = survey_final_gis)
summary(fit_pasture_log)
```

```
##
## Call:
## glm(formula = pasture_management ~ log(drainage_area_km), family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.16275    0.06229   2.613  0.00898 **
## log(drainage_area_km) 0.09512    0.02396   3.971 7.17e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1831.7  on 1339  degrees of freedom
## Residual deviance: 1815.5  on 1338  degrees of freedom
## AIC: 1819.5
##
## Number of Fisher Scoring iterations: 4
```

Similar results, but with a super low p for log(drainage area)

Next lets look at forest conservation

```
fit_forest <- glm(forest_conservation ~ drainage_area_km, family = binomial, data = survey_final_gis)
summary(fit_forest)
```

```
##
## Call:
## glm(formula = forest_conservation ~ drainage_area_km, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.4593884   0.0709507  20.569   <2e-16 ***
## drainage_area_km 0.0001294   0.0001311   0.987    0.324
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1289.6  on 1339  degrees of freedom
## Residual deviance: 1288.3  on 1338  degrees of freedom
## AIC: 1292.3
##
## Number of Fisher Scoring iterations: 5
```

```
fit_forest_log <- glm(forest_conservation ~ log(drainage_area_km), family = binomial, data = survey_final_gis)
summary(fit_forest_log)
```

```
##
## Call:
## glm(formula = forest_conservation ~ log(drainage_area_km), family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.30277    0.07451  17.485   < 2e-16 ***
## log(drainage_area_km) 0.17455    0.03290   5.305 1.12e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1289.6  on 1339  degrees of freedom
## Residual deviance: 1259.1  on 1338  degrees of freedom
## AIC: 1263.1
##
## Number of Fisher Scoring iterations: 4
```

Log(drainage area) is highly significant again

Let's now look at water management

```
fit_water <- glm(water_management ~ drainage_area_km, family = binomial, data = survey_final_gis)
summary(fit_water)
```

```
##
## Call:
## glm(formula = water_management ~ drainage_area_km, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.785e+00  7.901e-02  22.598  <2e-16 ***
## drainage_area_km 8.384e-05  1.268e-04   0.661   0.509
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1097.6  on 1339  degrees of freedom
## Residual deviance: 1097.0  on 1338  degrees of freedom
## AIC: 1101
##
## Number of Fisher Scoring iterations: 4
```

```
fit_water_log <- glm(water_management ~ log(drainage_area_km), family = binomial, data = survey_final_gis)
summary(fit_water_log)
```

```
##
## Call:
## glm(formula = water_management ~ log(drainage_area_km), family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.71958    0.08552  20.108  <2e-16 ***
## log(drainage_area_km) 0.06572    0.03416   1.924   0.0543 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1097.6  on 1339  degrees of freedom
## Residual deviance: 1093.8  on 1338  degrees of freedom
## AIC: 1097.8
##
## Number of Fisher Scoring iterations: 4
```

Similar results again. Log(drainage area) is significant for each of the adaptation categories.

What happens if I filter by “have cattle” and look at cattle_management

```
survey_havecattle <- survey_final_gis %>% filter(havecattle ==1)
fit_cattle_have <- glm(cattle_management ~ drainage_area_km, family = binomial, data = survey_havecattle)
summary(fit_cattle_have)
```

```
##
## Call:
```



```
## glm(formula = cattle_management ~ drainage_area_km, family = binomial,
##      data = survey_havecattle)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.137e-01  6.405e-02  -1.775   0.0759 .
## drainage_area_km  1.082e-04  8.377e-05   1.291   0.1966
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1389.3  on 1003  degrees of freedom
## Residual deviance: 1387.4  on 1002  degrees of freedom
## AIC: 1391.4
##
## Number of Fisher Scoring iterations: 3
```

```
fit_cattle_have_log <- glm(cattle_management ~ log(drainage_area_km), family = binomial, data = survey_h
summary(fit_cattle_have_log)
```

```
##
## Call:
## glm(formula = cattle_management ~ log(drainage_area_km), family = binomial,
##      data = survey_havecattle)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.13422    0.07437  -1.805   0.0711 .
## log(drainage_area_km)  0.02411    0.02729   0.883   0.3770
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1389.3  on 1003  degrees of freedom
## Residual deviance: 1388.6  on 1002  degrees of freedom
## AIC: 1392.6
##
## Number of Fisher Scoring iterations: 3
```

Interesting, if we only include those that have cattle, the significance of the log(drainage area) disappears.

Let's see if anything similar happens when filtered by havecattle for any of the other adaptations

```
fit_pasture_have_log <- glm(pasture_management ~ log(drainage_area_km), family = binomial, data = survey_h
summary(fit_pasture_have_log)
```

```
##
## Call:
## glm(formula = pasture_management ~ log(drainage_area_km), family = binomial,
##      data = survey_havecattle)
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.76591    0.07953   9.630  <2e-16 ***
## log(drainage_area_km) 0.07154    0.03075   2.327    0.02 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 1221.1  on 1003  degrees of freedom
## Residual deviance: 1215.6  on 1002  degrees of freedom
## AIC: 1219.6
##
## Number of Fisher Scoring iterations: 4
```

```
fit_forest_have_log <- glm(forest_conservation ~ log(drainage_area_km), family = binomial, data = survey_hav
summary(fit_forest_have_log)
```

```
##
## Call:
## glm(formula = forest_conservation ~ log(drainage_area_km), family = binomial,
##      data = survey_havecat)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.22569    0.12126  18.354  < 2e-16 ***
## log(drainage_area_km) 0.26151    0.06041   4.329  1.5e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 553.3  on 1003  degrees of freedom
## Residual deviance: 532.1  on 1002  degrees of freedom
## AIC: 536.1
##
## Number of Fisher Scoring iterations: 6
```

```
fit_water_have_log <- glm(water_management ~ log(drainage_area_km), family = binomial, data = survey_hav
summary(fit_forest_have_log)
```

```
##
## Call:
## glm(formula = forest_conservation ~ log(drainage_area_km), family = binomial,
##      data = survey_havecat)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.22569    0.12126  18.354  < 2e-16 ***
## log(drainage_area_km) 0.26151    0.06041   4.329  1.5e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 553.3  on 1003  degrees of freedom
## Residual deviance: 532.1  on 1002  degrees of freedom
## AIC: 536.1
##
## Number of Fisher Scoring iterations: 6
```

We still have a significant log(drainage area) for all of the other adaptation categories.

I'm curious about how many farms made multiple adaptations

```
table(survey_final_gis$cattle_management, survey_final_gis$pasture_management)
```

```
##
##      0  1
## 0 467 373
## 1 110 390
```

```
table(survey_final_gis$cattle_management, survey_final_gis$forest_conservation)
```

```
##
##      0  1
## 0 216 624
## 1  34 466
```

```
table(survey_final_gis$cattle_management, survey_final_gis$water_management)
```

```
##
##      0  1
## 0 169 671
## 1  22 478
```

```
table(survey_final_gis$pasture_management, survey_final_gis$forest_conservation)
```

```
##
##      0  1
## 0 194 383
## 1  56 707
```

```
table(survey_final_gis$pasture_management, survey_final_gis$water_management)
```

```
##
##      0  1
## 0 162 415
## 1  29 734
```

```
table(survey_final_gis$forest_conservation, survey_final_gis$water_management)
```

```
##
##      0      1
##    0 149 101
##    1  42 1048
```

It looks like the majority of farmers who adapted did more than one thing.

Let's also look at deforestation as a linear model

```
fit_deforest <- lm(cleared_area ~ drainage_area_km, data = survey_final_gis)
summary(fit_deforest)
```

```
##
## Call:
## lm(formula = cleared_area ~ drainage_area_km, data = survey_final_gis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -375.73  -55.97  -34.22    6.90  2102.51
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   75.319662   4.286540  17.571 < 2e-16 ***
## drainage_area_km 0.033203   0.004989   6.655 4.29e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 146.6 on 1195 degrees of freedom
## (143 observations deleted due to missingness)
## Multiple R-squared:  0.03574,    Adjusted R-squared:  0.03493
## F-statistic: 44.29 on 1 and 1195 DF,  p-value: 4.295e-11
```

```
fit_deforest_log <- lm(cleared_area ~ log(drainage_area_km), data = survey_final_gis)
summary(fit_deforest_log)
```

```
##
## Call:
## lm(formula = cleared_area ~ log(drainage_area_km), data = survey_final_gis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -172.64  -52.52  -24.31    7.40  2093.49
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    59.849     4.797  12.475 <2e-16 ***
## log(drainage_area_km) 14.910     1.759   8.477 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 145 on 1195 degrees of freedom
## (143 observations deleted due to missingness)
## Multiple R-squared:  0.05672,    Adjusted R-squared:  0.05594
## F-statistic: 71.86 on 1 and 1195 DF,  p-value: < 2.2e-16
```

We see significance for both the drainage area and the log(drainage area). Both had positive coefficients, which is confusing if my understanding of drainage_area and cleared_area are both correct.

Look at models with vehicle value and drainage area. Drainage area was not significant in any models, but log(drainage area) is for some.

```
survey_final_gis_wealth <- survey_final_gis %>% filter(!is.na(survey_final_gis$vechval))
fit_wealth_ldrain_water <- glm(water_management~ vechval + log(drainage_area_km), family = binomial, data = survey_final_gis_wealth)
fit_wealth_ldrain_cattle <- glm(cattle_management~ vechval + log(drainage_area_km), family = binomial, data = survey_final_gis_wealth)
fit_wealth_ldrain_pasture <- glm(pasture_management~ vechval + log(drainage_area_km), family = binomial, data = survey_final_gis_wealth)
fit_wealth_ldrain_forest <- glm(forest_conservation~ vechval + log(drainage_area_km), family = binomial, data = survey_final_gis_wealth)
#fit_wealth_drainn_any_adaptation <- glm(any_adaptation~ vechval + log(drainage_area_km), family = binomial, data = survey_final_gis_wealth)
summary(fit_wealth_ldrain_water)
```

```
##
## Call:
## glm(formula = water_management ~ vechval + log(drainage_area_km),
##      family = binomial, data = survey_final_gis_wealth)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      2.921e+00  1.999e-01  14.616  <2e-16 ***
## vechval           7.799e-06  3.710e-06   2.102  0.0355 *
## log(drainage_area_km) -7.259e-02  5.769e-02  -1.258  0.2083
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 421.94  on 1199  degrees of freedom
## Residual deviance: 414.38  on 1197  degrees of freedom
## AIC: 420.38
##
## Number of Fisher Scoring iterations: 7
```

```
summary(fit_wealth_ldrain_cattle)
```

```
##
## Call:
## glm(formula = cattle_management ~ vechval + log(drainage_area_km),
##      family = binomial, data = survey_final_gis_wealth)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -6.776e-01  8.061e-02  -8.407  < 2e-16 ***
## vechval           6.768e-06  1.014e-06   6.672 2.52e-11 ***
## log(drainage_area_km)  4.919e-03  2.559e-02   0.192   0.848
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 1630.1  on 1199  degrees of freedom
```

```
## Residual deviance: 1571.4 on 1197 degrees of freedom
## AIC: 1577.4
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_ldrain_pasture)
```

```
##
## Call:
## glm(formula = pasture_management ~ vechval + log(drainage_area_km),
##      family = binomial, data = survey_final_gis_wealth)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.764e-01  8.196e-02   2.152  0.0314 *
## vechval         7.178e-06  1.293e-06   5.553 2.81e-08 ***
## log(drainage_area_km) 5.432e-02  2.627e-02   2.068  0.0387 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1573.9 on 1199 degrees of freedom
## Residual deviance: 1521.6 on 1197 degrees of freedom
## AIC: 1527.6
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_ldrain_forest)
```

```
##
## Call:
## glm(formula = forest_conservation ~ vechval + log(drainage_area_km),
##      family = binomial, data = survey_final_gis_wealth)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.938e+00  1.266e-01  15.313 < 2e-16 ***
## vechval         4.393e-06  2.241e-06   1.960  0.05 *
## log(drainage_area_km) 2.058e-01  4.851e-02   4.242 2.22e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 735.30 on 1199 degrees of freedom
## Residual deviance: 706.36 on 1197 degrees of freedom
## AIC: 712.36
##
## Number of Fisher Scoring iterations: 6
```

```
#summary(fit_wealth_drain_any_adaptation )
```

The model for any adaptation did not converge, but all of the others did. Vehicle value was still significant for all of the adaptation measures. Log(drainage area) was not significant for water or cattle management when vehicle value is included, but it is for pasture management and forest conservation. For forest conservation, drainage area has a much smaller p value than vehicle value.

Try using draianage area with the tail chopped off instead of log transform

```
survey_final_gis_wealth_filtered <- survey_final_gis_wealth %>% filter(drainage_area_km < 500)
fit_wealth_drain_water <- glm(water_management~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_cattle <- glm(cattle_management~ vechval + drainage_area_km , family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_pasture <- glm(pasture_management~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_forest <- glm(forest_conservation~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
#fit_wealth_drainn_any_adaptation <- glm(any_adaptation~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
summary(fit_wealth_drain_water)
```

```
##
## Call:
## glm(formula = water_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.881e+00  1.920e-01  15.004  <2e-16 ***
## vechval         6.662e-06  3.628e-06   1.836  0.0663 .
## drainage_area_km -1.470e-03  2.511e-03  -0.586  0.5582
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 404.88  on 1146  degrees of freedom
## Residual deviance: 399.83  on 1144  degrees of freedom
## AIC: 405.83
##
## Number of Fisher Scoring iterations: 6
```

```
summary(fit_wealth_drain_cattle)
```

```
##
## Call:
## glm(formula = cattle_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.819e-01  8.071e-02  -8.449  < 2e-16 ***
## vechval        6.748e-06  1.044e-06   6.463 1.03e-10 ***
## drainage_area_km 1.257e-04  1.212e-03   0.104   0.917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1555.4 on 1146 degrees of freedom
## Residual deviance: 1503.0 on 1144 degrees of freedom
## AIC: 1509
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_drain_pasture)
```

```
##
## Call:
## glm(formula = pasture_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.298e-01  8.193e-02   2.804  0.00504 **
## vechval         7.084e-06  1.300e-06   5.449 5.07e-08 ***
## drainage_area_km 2.042e-04  1.251e-03   0.163  0.87040
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1509.1 on 1146 degrees of freedom
## Residual deviance: 1468.3 on 1144 degrees of freedom
## AIC: 1474.3
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_drain_forest)
```

```
##
## Call:
## glm(formula = forest_conservation ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.958e+00  1.345e-01  14.557  <2e-16 ***
## vechval         5.166e-06  2.287e-06   2.259  0.0239 *
## drainage_area_km 1.018e-02  4.895e-03   2.080  0.0376 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 702.18 on 1146 degrees of freedom
## Residual deviance: 686.66 on 1144 degrees of freedom
## AIC: 692.66
##
## Number of Fisher Scoring iterations: 6
```



```
#summary(fit_wealth_drain_any_adaptation )
```

With the drainage area outliers removed, drainage area is significant for forest conservation but none of the others.

```
rainfall_group<- cut(survey_final_gis_wealth_filtered$rainfall , c(0,1950,2300), label=c("low","high"))
rainfall_group <-factor(rainfall_group)
survey_final_gis_wealth_filtered$studycode <-factor(survey_final_gis_wealth_filtered$studycode)
fit_wealth_drain_water <- glm(water_management~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_cattle <- glm(cattle_management~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_pasture <- glm(pasture_management~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
fit_wealth_drain_forest <- glm(forest_conservation~ vechval + log(drainage_area_km) + log(rainfall_yearly), family = binomial, data = survey_final_gis_wealth_filtered)
#fit_wealth_drainn_any_adaptation <- glm(any_adaptation~ vechval + drainage_area_km, family = binomial, data = survey_final_gis_wealth_filtered)
summary(fit_wealth_drain_water)
```

```
##
## Call:
## glm(formula = water_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.881e+00  1.920e-01  15.004  <2e-16 ***
## vechval         6.662e-06  3.628e-06   1.836  0.0663 .
## drainage_area_km -1.470e-03  2.511e-03  -0.586  0.5582
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 404.88  on 1146  degrees of freedom
## Residual deviance: 399.83  on 1144  degrees of freedom
## AIC: 405.83
##
## Number of Fisher Scoring iterations: 6
```

```
summary(fit_wealth_drain_cattle)
```

```
##
## Call:
## glm(formula = cattle_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.819e-01  8.071e-02  -8.449  < 2e-16 ***
## vechval        6.748e-06  1.044e-06   6.463 1.03e-10 ***
## drainage_area_km 1.257e-04  1.212e-03   0.104   0.917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
## Null deviance: 1555.4 on 1146 degrees of freedom
## Residual deviance: 1503.0 on 1144 degrees of freedom
## AIC: 1509
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_drain_pasture)
```

```
##
## Call:
## glm(formula = pasture_management ~ vechval + drainage_area_km,
##      family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.298e-01  8.193e-02   2.804  0.00504 **
## vechval         7.084e-06  1.300e-06   5.449 5.07e-08 ***
## drainage_area_km 2.042e-04  1.251e-03   0.163  0.87040
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1509.1 on 1146 degrees of freedom
## Residual deviance: 1468.3 on 1144 degrees of freedom
## AIC: 1474.3
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit_wealth_drain_forest)
```

```
##
## Call:
## glm(formula = forest_conservation ~ vechval + log(drainage_area_km) +
##      log(rainfall_year) + studycode, family = binomial, data = survey_final_gis_wealth_filtered)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    6.459e+00  3.104e+01   0.208   0.835
## vechval         3.364e-06  2.276e-06   1.478   0.139
## log(drainage_area_km)  2.767e-01  5.549e-02   4.987 6.14e-07 ***
## log(rainfall_year)    -5.694e-01  4.011e+00  -0.142   0.887
## studycodeOuro Preto do Oeste -2.466e-01  3.167e-01  -0.779   0.436
## studycodeRolim de Moura      4.268e-02  6.275e-01   0.068   0.946
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 702.18 on 1146 degrees of freedom
## Residual deviance: 665.23 on 1141 degrees of freedom
## AIC: 677.23
```

```
##
## Number of Fisher Scoring iterations: 6
```

```
#summary(fit_wealth_drain_any_adaptation )
```

Since we've seen that wealth is so important for adaptation, but drainage area importance decreases for wealth, let's see if other water availability proxies are significant for the individual adaptations

```
fit_cattle_full <- glm(cattle_management ~ SPImax_year + vechval, family = binomial, data = survey_final_gis)
summary(fit_cattle)
```

```
##
## Call:
## glm(formula = cattle_management ~ drainage_area_km, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -5.298e-01  5.717e-02  -9.267   <2e-16 ***
## drainage_area_km  8.492e-05  6.657e-05   1.276    0.202
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1770.4  on 1339  degrees of freedom
## Residual deviance: 1768.7  on 1338  degrees of freedom
## AIC: 1772.7
##
## Number of Fisher Scoring iterations: 4
```

```
fit_water_full <- glm(water_management ~ SPImax_year + vechval, family = binomial, data = survey_final_gis)
summary(fit_water_full)
```

```
##
## Call:
## glm(formula = water_management ~ SPImax_year + vechval, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.223e+00  5.507e-01   7.669 1.73e-14 ***
## SPImax_year -9.839e-01  3.508e-01  -2.805  0.00503 **
## vechval      7.471e-06  3.777e-06   1.978  0.04792 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 421.94  on 1199  degrees of freedom
## Residual deviance: 408.44  on 1197  degrees of freedom
## (140 observations deleted due to missingness)
```

```

## AIC: 414.44
##
## Number of Fisher Scoring iterations: 7

fit_pasture_full <- glm(pasture_management ~ rainfall + vechval, family = binomial, data = survey_final_gis)
summary(fit_pasture_full)

##
## Call:
## glm(formula = pasture_management ~ rainfall + vechval, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.909e+00  1.071e+00   2.715  0.00663 **
## rainfall     -1.345e-03  5.365e-04  -2.507  0.01217 *
## vechval       7.361e-06  1.299e-06   5.668  1.45e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1573.9  on 1199  degrees of freedom
## Residual deviance: 1519.6  on 1197  degrees of freedom
##      (140 observations deleted due to missingness)
## AIC: 1525.6
##
## Number of Fisher Scoring iterations: 4

fit_forest_full <- glm(forest_conservation ~ SPImin_year + vechval, family = binomial, data = survey_final_gis)
summary(fit_forest_full)

##
## Call:
## glm(formula = forest_conservation ~ SPImin_year + vechval, family = binomial,
##      data = survey_final_gis)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.165e+00  4.268e-01   2.729  0.00634 **
## SPImin_year  -8.688e-01  4.047e-01  -2.147  0.03183 *
## vechval       5.614e-06  2.248e-06   2.497  0.01252 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 735.30  on 1199  degrees of freedom
## Residual deviance: 721.45  on 1197  degrees of freedom
##      (140 observations deleted due to missingness)
## AIC: 727.45
##
## Number of Fisher Scoring iterations: 6

```