Lab 13

Previous data wrangling:

```r
GripStrengthMeasurements <- read_excel("GripStrengthMeasurementsS25.xlsx")
GS <- GripStrengthMeasurements %>% drop_na(GripStrength) #Mostly cleans out
empty rows

GS2 <- GS %>% drop_na(Arm)
GS2 <- GS2 %>% mutate(Arm = forcats::fct_collapse(Arm,
                                                Up = c("Up", "up"),
                                                Down = c("Down", "down"),
                                                "90" = c("90", "90 degree")),
                    SubjectID = forcats::fct_collapse(SubjectID,
                                                MDEarl = c("MDEarl","MDEar")),
                    Arm = factor(Arm),
                    Hand = factor(Hand),
                    OrderF = factor(Order)
)

GS2 <- GS2 %>% mutate(Hand.Arm = factor(str_c(Hand, Arm)))

GS2 <- GS %>% drop_na(Arm)
GS2 <- GS2 %>% mutate(Arm = forcats::fct_collapse(Arm,
                                                Up = c("Up", "up"),
                                                Down = c("Down", "down"),
                                                "90" = c("90", "90 degree")),
                    SubjectID = forcats::fct_collapse(SubjectID,
                                                MDEarl = c("MDEarl","MDEar")),
                    Arm = factor(Arm),
                    Hand = factor(Hand),
                    OrderF = factor(Order)
)
Demographics <- read_excel("DemographicsS25.xlsx")
Demographics <- Demographics %>% dplyr::select(-c(12:13)) #Because of a
"note" left in column 13...
Demographics <- Demographics %>% mutate(SubjectID =
                forcats::fct_recode(SubjectID,
                    LADeathGrips = "LADeath",
                    CenntenialWarriors = "CentennialWarriors")
)

Demographics <- Demographics %>% mutate(Weights = factor(Weights),
                                    GripTesterID = factor(GripTesterID))

Demographics <- Demographics %>% mutate(Weights = fct_collapse(Weights,
no = c("no",  "No"),
```

```
              yes = c("yes", "Yes")))

#Join Demographics to the GS2 repeated measures data
combined <- left_join(x = GS2, y = Demographics, by = "SubjectID")

combinedR <- combined %>% drop_na()

combinedR <- combinedR %>% mutate(forearm_bins   =
                                    factor(cut_number(ForearmLength, n = 3)),
                                  epworth_bins =
                                    factor(cut_number(Epworth, n = 3)),
                                  balance_bins =
                                    factor(cut_number(BalanceTime, n = 3))
                                  )
combinedR <- combinedR %>% mutate(
              forearm_bins = fct_recode(forearm_bins,
                                        low = "[9.25,10]",
                                        medium = "(10,11]",
                                        high = "(11,13.1]")
              )
lmer3 <- lmer(GripStrength ~ Hand*Arm + forearm_bins + GripTesterID +
(1|SubjectID), data = combinedR)
```

Part I: Final Grip Strength model: Three-level model (from the end of lab 12)

1) The previous model for grip strength failed to account for groups and that you were nested into groups for taking measurements that might also be systematically different on grip strength. Modify lmer3 to fit a new model account for group and then make a new model_diagram and generate an Anova set of F-tests for the model. No discussion.

```
lmer4 <- lmer(GripStrength ~ Hand*Arm + forearm_bins + GripTesterID +
(1|GroupName/SubjectID), data = combinedR)

Anova(lmer4, test.statistic = "F")

## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
##
## Response: GripStrength
##                    F Df  Df.res    Pr(>F)
## Hand          19.5002  1 209.006 1.613e-05
## Arm            6.5154  2 209.006 0.0017994
## forearm_bins 10.1805  2  33.920 0.0003443
## GripTesterID  1.3293  4   5.739 0.3625960
## Hand:Arm       0.5782  2 209.006 0.5618222

model_diagram(lmer4, heightVal = 800)
```
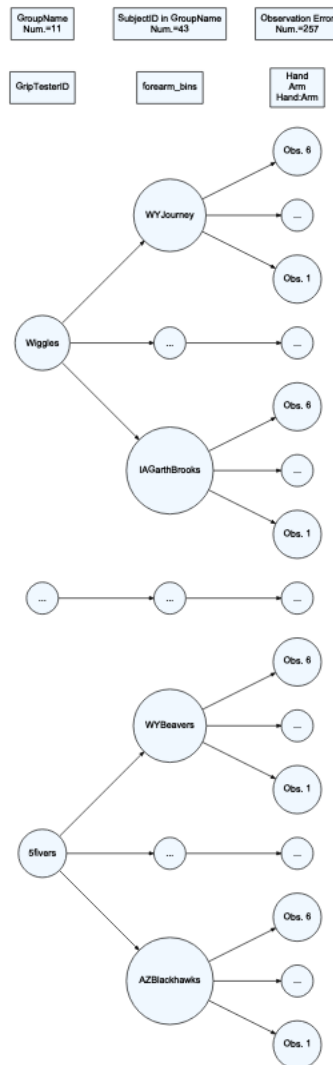
2) Discuss how the denominator DF from the F-tests loosely/qualitatively relate to the `model_diagram` results for the Grip Tester, forearm, and Hand fixed effects. Report how many unique values were present at each level (provided in the model diagram) and the denominator DF in the discussion. Note that this is not about a formula but about relative size and number of unique observations at each level, *since the denominator DF can't be more than the number of unique observations at each level*.

The model diagram shows 11 groups, 43 subjects, and 257 observations. The denominator DF generally make sense because the griptester DF is 5.739, forearm_bins is 33.920, and hand, arm, hand:arm all have denominator DF of 209.006. These relative sizes line up with the unique values in our model diagram for the levels where these groups were applied.

3) Generate a caterpillar plot from your new model. Find your group or a group of interest to you. Discuss the relative location of that group and, if you know something

about the group, discuss whether this result is something you expected given how we formed the groups based on height of the students.

At the group level, the TallerThanYou group had relatively higher estimated mean grip strength than the rest of the groups, and the average. This makes sense as these individuals were the tallest in their class last semester, and included PATaller who had the highest grip strength of all.

```r
ranef(lmer4)
```

```
## $`SubjectID:GroupName`
##                                (Intercept)
## ABH1997:BANAVIG                 -7.5761812
## AFEDU2020:BANAVIG              -14.0182654
## AustinLions:GentleGiants       -26.4846160
## AZBlackhawks:5fivers             2.9398201
## AZRams:5fivers                  -2.5804367
## BelknapBull:Shawties             2.7620350
## BostonLakers:5fivers            12.6558283
## BostonWolverines:GentleGiants    9.6406761
## CAJustinBieber:Boxplots         -6.9782713
## CenntenialWarriors:Shawties     -0.7693745
## COMichealFranti:Snuggies         8.2861776
## DCSNAP:SNAPCREW                 -3.6384098
## DenverCoyotes:Shawties          -0.7592028
## DenverGalaxy:BANAVIG            11.8884391
## FLTaller:TallerThanYou          -7.6098964
## GASNAP:SNAPCREW                  4.8521628
## HYENA:BESTBY                    25.7167890
## IAGarthBrooks:Wiggles           12.5002301
## IDAJR:GrapeJuice                -7.2854334
## IDPAD:GrapeJuice                 6.3532329
## LACROIX:BESTBY                 -36.1847913
## LADeathGrips:Snuggies          -12.6273592
## NDLawrence:Wiggles              -0.7696199
## NewYorkLaeeb:BANAVIG            -9.8808974
## NMSnuggies:Snuggies              2.4384973
## NVKP:GrapeJuice                 -5.1335659
## OrlandoChamp:GentleGiants       10.5861936
## ORMichelangelo:Boxplots          0.5858693
## PATaller:TallerThanYou          23.3673020
## SanDiegoBobcat:Shawties         -6.3671002
## UTSNAP:SNAPCREW                -12.4089004
## VancouverMcLaren:BESTBY         11.7724255
## VTPink:Boxplots                 -8.2172253
## WATaller:TallerThanYou          -9.8045397
## WinonaBadgers:GentleGiants       1.2968216
## WIWiggles:Wiggles               12.4541075
## WYBeavers:5fivers              -11.1125272
## WYIW:GrapeJuice                  2.1799140
```

```
## WYJourney:Wiggles              -4.5978128
## WYSlipKnot:Boxplots             3.3898180
## WYSNAP:SNAPCREW                 16.3287899
## WYTaller:TallerThanYou          10.2278681
## ZOO:BESTBY                       2.5814291
##
## $GroupName
##              (Intercept)
## 5fivers         1.148931
## BANAVIG       -11.827506
## BESTBY          2.346463
## Boxplots       -6.775055
## GentleGiants   -2.995643
## GrapeJuice     -2.346463
## Shawties       -3.099938
## SNAPCREW        3.099938
## Snuggies       -1.148931
## TallerThanYou   9.770698
## Wiggles        11.827506
##
## with conditional variances for "SubjectID:GroupName" "GroupName"
```

```r
dotplot(ranef(lmer4, postVar = T))
```

```
## $`SubjectID:GroupName`
```



```
##
## $GroupName
```
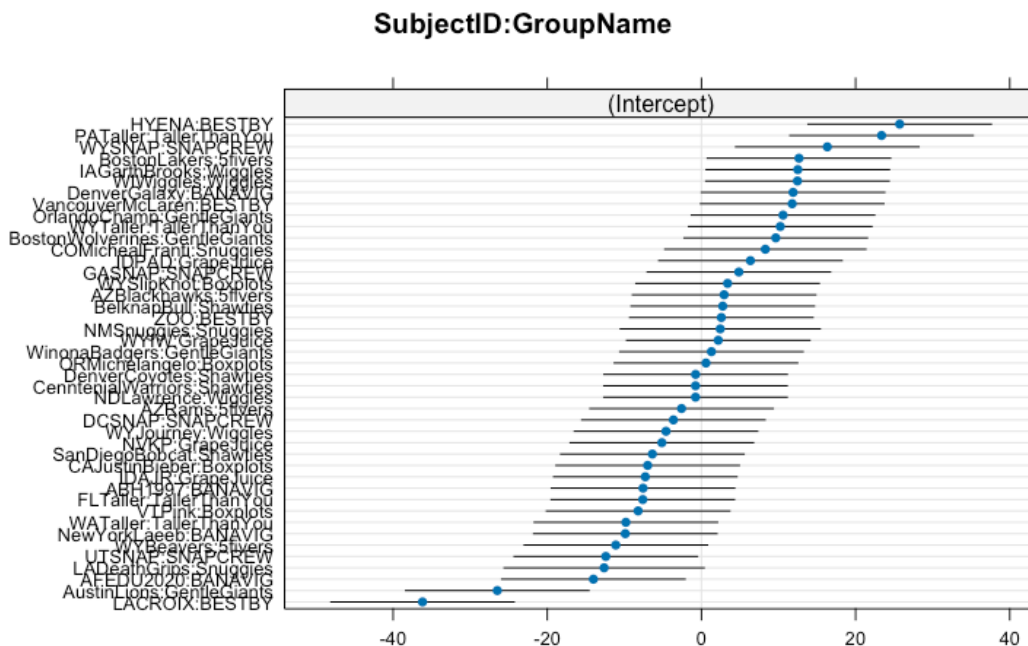
GroupName
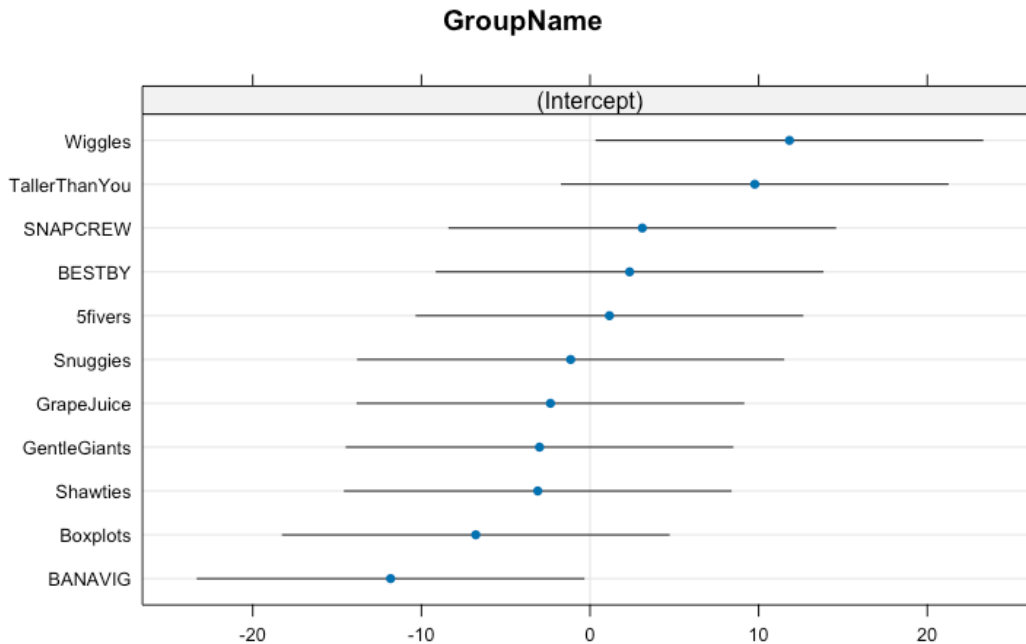
Part II: Complex field study multi-level model

- Read Uselman et al. (2018) and focus on the study design (Figure 2) and the results for the "South" field for Shrub Density (Table 2) and their 3-factor model. We won't consider all the interactions of the fixed effects and will handle some of the variables slightly differently below.

- Uselman, SM, Davison, J, Baughman, OW, Sullivan, BW, Miller, WW, and EA Leger (2018) Restoring dryland old fields with native shrubs and grasses: Does facilitation and seed source matter? *PLoS ONE* 13(10): e0205760. https://doi.org/10.1371/journal.pone.0205760

```r
Uselman <- read_csv("Uselman_data.csv", na = "-")
UselmanR <- Uselman %>% dplyr::filter(Site == "South") %>%
  mutate(Date = factor(Date)) %>%
  dplyr::filter(Date == "16-May") %>%
  mutate(SeedingStrat = factor(SeedingStrat),
         Irrigation = factor(Irrigation),
         GrassOrig = factor(GrassOrig),
         Blockf = factor(Block),
         logShrubDens25 = log(ShrubDens2016 + 0.025),
         Block = factor(Block),
         SubBlock = factor(SubBlock),
         Plot = factor(Plot))

UselmanR <- UselmanR %>%  mutate(SeedingStrat = factor(SeedingStrat),
         Irrigation = factor(Irrigation),
         GrassOrig = factor(GrassOrig),
```

```
        SeedingStrat.ShrubOrig = factor(str_c(SeedingStrat, ShrubOrig)),
        SeedingStrat.ShrubOrig =
forcats::fct_na_value_to_level(SeedingStrat.ShrubOrig, "V")
        )
```

4) Work on the whiteboards to develop and write out a model diagram for the study design described in Figure 2 with predictors of just Seeding Strategy and Irrigation. The diagram will not be graded for correctness, but take a picture of the diagram you develop and read the file into the document.

- You can read image files into Rmd's by placing them in the same location as the Rmd and using something like `![Figure 1. Preliminary Model Diagram](mydiagram.png)` on a separate line in the file and not within a codechunk, depending on the type of image file you generate.
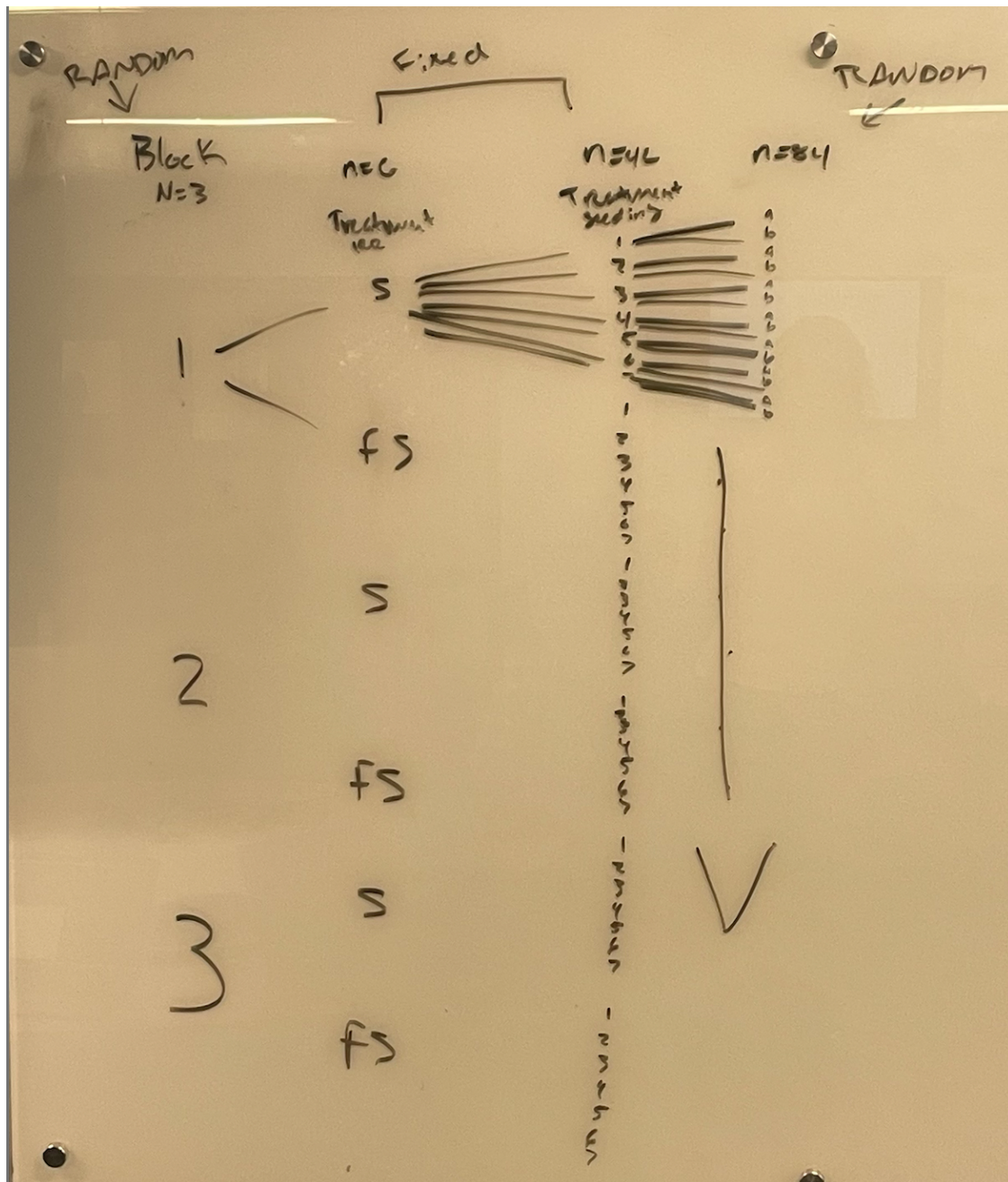
Figure 1. Preliminary Model Diagram

5) What is Seeding Strategy level "V" in the study design?

Seeding Strategy level "V" is the seeding control where they did not seed in anything.

6) The authors used shrub origin and seeding strategy and their interaction (among other things) in their three-factor model in Table 2. Report their test statistic and distribution under the null for the seeding strategy by shrub orig interaction in that three-factor model. What do the numerator DF in the table tell you about the number of levels for seeding strategy and shrub origin in this model?
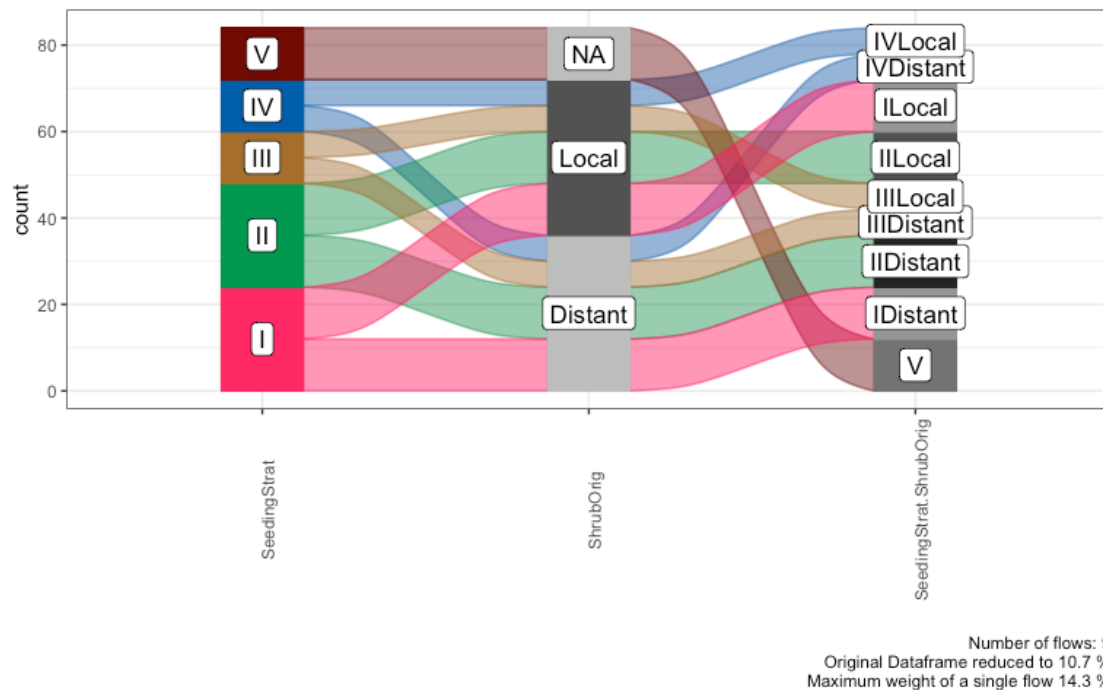
$F_{3,28} = 10.80$, $p-value < 0.001$

The numerator df would say that there are 4 seeding strategies being used for the 2 different shrub origins.

7) The code above creates a variable called `SeadingStrat.ShrubOrig`. Explain the levels of SeedinStrat.ShrubOrig, especially as it relates to level Seeding Strategy Level V and why there were missing data on the shrub origin variable.

SeedinStrat.ShrubOrig creates levels that are the combination of the seeding strategy and the shrub origin. So for example strategy IV would have two levels in this variable called IVLocal and IVDistant. Seeding strategy V is the control and there is no seeding being done, so it does not apply in the new variable as there is no seeding of shrubs to combine.

```
UselmanR %>%
  dplyr::select(SeedingStrat, ShrubOrig, SeedingStrat.ShrubOrig) %>%
  alluvial_wide()
```


Number of flows: 9
Original Dataframe reduced to 10.7 %
Maximum weight of a single flow 14.3 %

8) The following models attempt to fit models to account for the nested measurements in the study design with block, "subblocks" in blocks (shaded areas in their Figure 2), and plots in subblocks in blocks (the numbers 1 to 7 that are then split into A and B parts at the dashed line in their Figure 2 - more or less the rows within the subblocks). The models are fit two ways, one using the forced nesting notation and the other trying to have `lmer` find the nesting structure using the additive notation for the random effects. What changed between the two models in terms of the counts of levels in the random effects?

The counts in levels of random effects for lmerU4 differ from lmerU4A in the group count. lmerU4 is the forced nesting option and creates 42 separate groups for the plot:SubBlock:Block, whereas the lmerU4A only shows 7 that account for the different seeding strategies. lmerU4A then did not split the seeding strategy into random effect groups.

lmerU4: Number of obs: 84, groups: Plot:SubBlock:Block, 42; SubBlock:Block, 6; Block, 3
lmerU4A: Number of obs: 84, groups: Plot, 7; SubBlock, 6; Block, 3

```
lmerU4 <- lmer(logShrubDens25 ~ 1 + (1|Block/SubBlock/Plot), data =
UselmanR)

lmerU4A <- lmer(logShrubDens25 ~ 1 + (1|Block) + (1|SubBlock) + (1|Plot),
data = UselmanR)

summary(lmerU4)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: logShrubDens25 ~ 1 + (1 | Block/SubBlock/Plot)
##    Data: UselmanR
##
## REML criterion at convergence: 312.5
##
## Scaled residuals:
##     Min      1Q   Median      3Q      Max
## -1.70890 -0.60660 -0.07192  0.45771  1.85651
##
## Random effects:
##  Groups               Name        Variance Std.Dev.
##  Plot:SubBlock:Block (Intercept) 1.20211  1.0964
##  SubBlock:Block       (Intercept) 0.04413  0.2101
##  Block                (Intercept) 0.00000  0.0000
##  Residual                         1.47239  1.2134
## Number of obs: 84, groups:
## Plot:SubBlock:Block, 42; SubBlock:Block, 6; Block, 3
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  -1.6851     0.2313  5.0000  -7.285 0.000762
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')

summary(lmerU4A)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: logShrubDens25 ~ 1 + (1 | Block) + (1 | SubBlock) + (1 | Plot)
##    Data: UselmanR
##
```

```
## REML criterion at convergence: 293.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.31216 -0.62404 -0.06676  0.76433  1.66956
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  Plot     (Intercept) 1.155e+00 1.075e+00
##  SubBlock (Intercept) 2.128e-01 4.613e-01
##  Block    (Intercept) 1.999e-09 4.472e-05
##  Residual             1.516e+00 1.231e+00
## Number of obs: 84, groups:  Plot, 7; SubBlock, 6; Block, 3
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  -1.6851     0.4675  7.7480  -3.605  0.00731
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

9) It can be good to explore the random effect levels and combinations you are using with an alluvial diagram. Modify the provided code to make one of the `Block`, `SubBlock`, and `Plot`. No discussion.

```
UselmanR %>% dplyr::select(Block, SubBlock, Plot) %>% alluvial_wide(fill_by =
"first_variable")
```
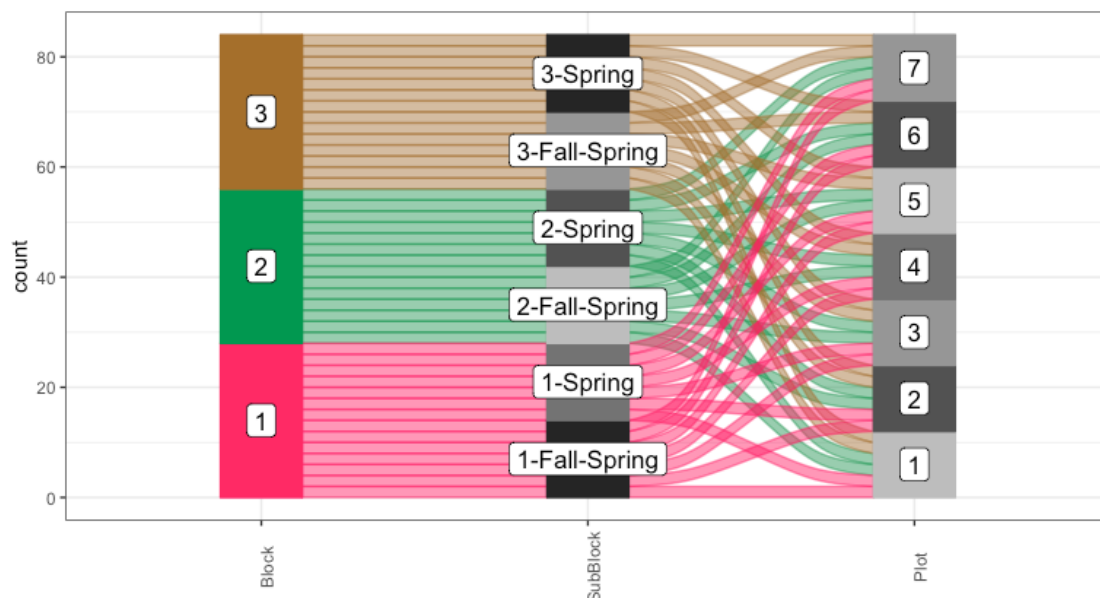

Number of flows: 42
Original Dataframe reduced to 50 %
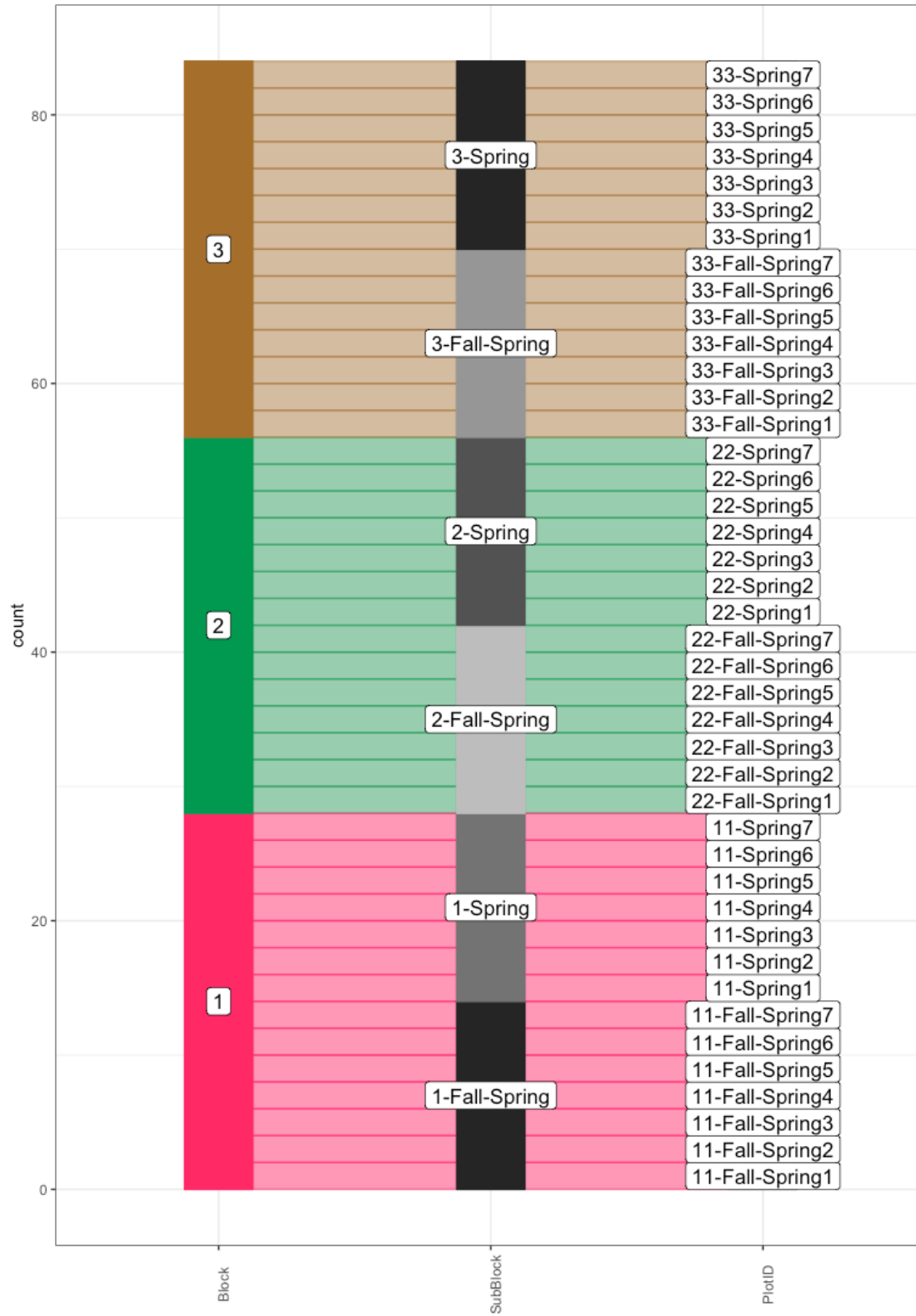Maximum weight of a single flow 2.4 %

10) The issue in the "additive" random effect model notation is that `Plot` reused the same coding of 1 to 7 for different locations within different blocks and plots. What is

needed is a unique code for each plot, something like `PlotID`, not a code that is generic and recycled across those higher level units. You can use the `str_c` function to fuse together the block, subblock, and plot information and create a new variable that uniquely identifies each plot. Replace `Plot` in the previous alluvial diagram and color the alluvia based on the new variable. How can you explain the "Number of flows" information in the alluvial diagram based on the study design?

We see 42 flows which makes sense since there are 42 separate "ways" for these variables to connect. Meaning, we have 3 blocks that are then split into 2 irrigation treatments leaving us with 6. These 6 are then split further into the 7 seeding types, leaving us with 42 flows.

```
UselmanR <- UselmanR %>% mutate(PlotID = str_c(Block, SubBlock, Plot))

UselmanR %>% dplyr::select(Block, SubBlock, PlotID) %>% alluvial_wide(fill_by
= "first_variable")
```

Number of flows: 42
Original Dataframe reduced to 50 %
Maximum weight of a single flow 2.4 %

11) Replace the `Plot` in the "additive" code version of the lmer model above with the uniquely coded version of the variable you created, `PlotID`. Also incorporate the fixed effects of the `SeedingStrat.ShrubOrig` variable which contains all the combinations of those two variables, `Irrigation`, and their interaction. How many levels does the new version of the "plot" that you created have in the model?

The PlotID level now has 42 levels!

```
lmerInt <- lmer(logShrubDens25 ~ SeedingStrat.ShrubOrig*Irrigation +
(1|Block) + (1|SubBlock) + (1|PlotID), data = UselmanR)

summary(lmerInt)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: logShrubDens25 ~ SeedingStrat.ShrubOrig * Irrigation + (1 |
Block) +
##      (1 | SubBlock) + (1 | PlotID)
##    Data: UselmanR
##
## REML criterion at convergence: 217
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4520 -0.4855 -0.0220  0.5676  1.6104
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  PlotID   (Intercept) 0.4632   0.6806
##  SubBlock (Intercept) 0.2763   0.5256
##  Block    (Intercept) 0.0000   0.0000
##  Residual             0.6240   0.7899
## Number of obs: 84, groups:  PlotID, 42; SubBlock, 6; Block, 3
##
## Fixed effects:
##                                              Estimate Std. Error
df
## (Intercept)                                   -1.4100     0.5228
16.4345
## SeedingStrat.ShrubOrigIIDistant               -0.6997     0.6020
48.5918
## SeedingStrat.ShrubOrigIIIDistant              -0.2079     0.7373
48.5918
## SeedingStrat.ShrubOrigIIILocal                 0.1418     0.7373
48.5918
## SeedingStrat.ShrubOrigIILocal                 -0.6714     0.6020
48.5918
## SeedingStrat.ShrubOrigILocal                   1.4444     0.4561
34.0000
## SeedingStrat.ShrubOrigIVDistant               -0.7760     0.7373
```

```
48.5918
## SeedingStrat.ShrubOrigIVLocal                               1.7192      0.7373
48.5918
## SeedingStrat.ShrubOrigV                                    -1.7547      0.6630
35.4907
## IrrigationSpring                                            0.3219      0.7393
16.4345
## SeedingStrat.ShrubOrigIIDistant:IrrigationSpring   -0.6901      0.8513
48.5918
## SeedingStrat.ShrubOrigIIIDistant:IrrigationSpring  -1.7776      1.0427
48.5918
## SeedingStrat.ShrubOrigIIILocal:IrrigationSpring    -2.1273      1.0427
48.5918
## SeedingStrat.ShrubOrigIILocal:IrrigationSpring     -0.6869      0.8513
48.5918
## SeedingStrat.ShrubOrigILocal:IrrigationSpring       0.5120      0.6450
34.0000
## SeedingStrat.ShrubOrigIVDistant:IrrigationSpring   -0.6333      1.0427
48.5918
## SeedingStrat.ShrubOrigIVLocal:IrrigationSpring     -0.1445      1.0427
48.5918
## SeedingStrat.ShrubOrigV:IrrigationSpring           -0.4130      0.9376
35.4907
##                                                      t value Pr(>|t|)
## (Intercept)                                           -2.697  0.01559
## SeedingStrat.ShrubOrigIIDistant                       -1.162  0.25077
## SeedingStrat.ShrubOrigIIIDistant                      -0.282  0.77912
## SeedingStrat.ShrubOrigIIILocal                         0.192  0.84829
## SeedingStrat.ShrubOrigIILocal                         -1.115  0.27022
## SeedingStrat.ShrubOrigILocal                           3.167  0.00324
## SeedingStrat.ShrubOrigIVDistant                       -1.053  0.29774
## SeedingStrat.ShrubOrigIVLocal                          2.332  0.02390
## SeedingStrat.ShrubOrigV                               -2.647  0.01205
## IrrigationSpring                                       0.435  0.66890
## SeedingStrat.ShrubOrigIIDistant:IrrigationSpring      -0.811  0.42155
## SeedingStrat.ShrubOrigIIIDistant:IrrigationSpring     -1.705  0.09461
## SeedingStrat.ShrubOrigIIILocal:IrrigationSpring       -2.040  0.04678
## SeedingStrat.ShrubOrigIILocal:IrrigationSpring        -0.807  0.42371
## SeedingStrat.ShrubOrigILocal:IrrigationSpring          0.794  0.43278
## SeedingStrat.ShrubOrigIVDistant:IrrigationSpring      -0.607  0.54640
## SeedingStrat.ShrubOrigIVLocal:IrrigationSpring        -0.139  0.89036
## SeedingStrat.ShrubOrigV:IrrigationSpring              -0.440  0.66225

## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

12) Fit the same model as in the previous question using the nested notation. Then make a model diagram from the nested random effects version of the model. Explain how the Block and SubBlock correspond to the information in their Figure 2 and the co-location of any fixed effects at those two levels of the hierarchy.

Our Block has n=3 and the subblock has n=6. The subblock corresponds to the fixed effect of irrigation, which has 2 levels as seen in their df in table 2. Our subblocks have 6, because they are accounting for the random effect of block.

```
lmerIntNest <- lmer(logShrubDens25 ~ SeedingStrat.ShrubOrig*Irrigation +
(1|Block/SubBlock/Plot), data = UselmanR)

model_diagram(lmerIntNest, heightVal = 800)
```
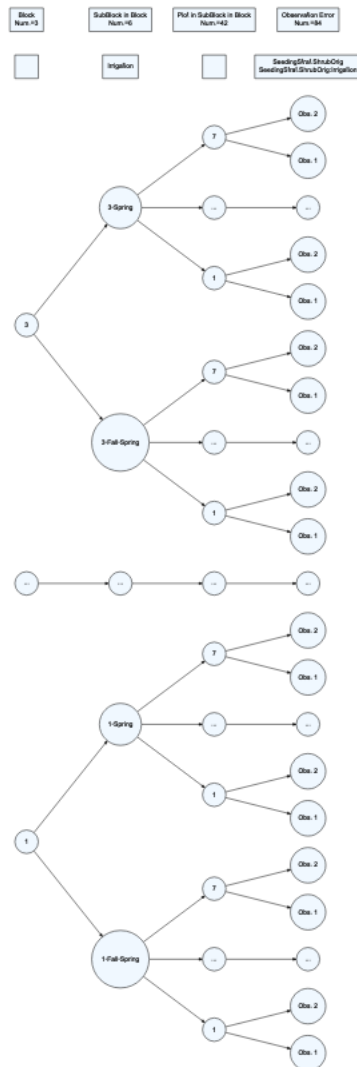

13) Note any additional resources used to complete this lab or NONE.

NONE