

Attracted by higher crude protein, grasshopper abundance and offtake increase after prescribed fire

Script

NG Heimbuch · DA McGranahan · CL Wonkka · L Vermeire · D Branson

2023-01-25

```
# Load packages
pacman::p_load(tidyverse, readxl, EnvStats)
# Load data
# Set path to main Excel file
fp = "../data/RangelandFireGrasshopperForage.xlsx"
# Treatments by plot
trt <- read_xlsx(fp, "Treatment")
# biomass data
fb <- read_xlsx(fp, "Forage Biomass") %>%
  full_join( trt , by="Plot")
# Forage quality data
cn<- read_xlsx(fp, "CN Analysis") %>%
  full_join( trt, by="Plot")
# Abundance data
ha<- read_xlsx(fp, "Hopper Abundance") %>%
  full_join(trt, by="Plot")
ring = 2.5 #sum area of five rings (square meters)

# Calculate offtake rate
off <-
fb %>%
  select(Treatment, Plot, Trial, Totalweight) %>%
  separate(Trial, into= c("Trt", "rep"), sep = " ") %>%
  mutate(rep = case_when(
    rep== "1"~"A",
    rep== "2"~"B",
    TRUE~ rep),
    Treatment = case_when(
      Treatment== "Summer"~"Unburned",
      TRUE~ Treatment),
    days= ifelse(rep== "A", 40, 33) ) %>%
  pivot_wider(names_from = Trt,
    values_from = Totalweight) %>%
  mutate(g_diff = (Control-Exclosure)/(Control), # offtake in grams/0.5 m^2 quadrat
    m_diff = g_diff * 2, # offtake in g/m^2
    kg_diff = m_diff * 10, # offtake in kg/ha (1 g/m^2 = 10 kg/ha)
    rate= kg_diff/days # daily kg/ha removed
  ) %>%
  group_by(Plot, Treatment) %>%
```

```

      summarize(rate = mean(rate),
                 .groups = 'drop')
#
# Stats
#
# 1. Test treatments against 0.
off_sum <-
  lm(rate ~ 0 + Treatment, off) %>%
  summary() %>%
  broom::tidy() %>%
  rename(p = p.value) %>%
  mutate(across(c(estimate, statistic), ~ as.character(signif(., 2))),
         p = case_when(
           p >= 0.05 ~ '> 0.05',
           between(p, 0.001, 0.009) ~ '< 0.01',
           p < 0.001 ~ '< 0.001',
           TRUE ~ paste0('= ', round(p, 2)) ))
# 2. Test treatments against each other (post-hoc pairwise comparison)
off_tuk <-
  lm(rate ~ Treatment, off) %>%
  aov() %>%
  TukeyHSD() %>%
  broom::tidy() %>%
  rename(p = adj.p.value) %>%
  mutate(estimate = paste0(signif(estimate, 2)),
         p = case_when(
           p >= 0.05 ~ '> 0.05',
           between(p, 0.001, 0.009) ~ '< 0.01',
           p < 0.001 ~ '< 0.001',
           TRUE ~ paste0('= ', round(p, 2)) ))

# Fit the protein test model
pmod <- lm(Protein~Treatment*Organ, cn)
# ANOVA
val_sum <-
  pmod %>%
  anova() %>%
  broom::tidy() %>%
  rename(p = p.value,
         Fstat = statistic) %>%
  mutate(Fstat = as.character(signif(Fstat, 2)),
         p = case_when(
           p >= 0.05 ~ '> 0.05',
           between(p, 0.001, 0.009) ~ '< 0.01',
           p < 0.001 ~ '< 0.001',
           TRUE ~ paste0('= ', round(p, 2)) ))
bd <- filter(cn, Treatment != "Summer")$Protein
burned_sum <- paste0(round(mean(bd), 1),
                    '% ± ',
                    round((sd(bd)/sqrt(length(bd))), 1),
                    ' s.e.')
# Post-hoc TUkey test on treatment pairwise comparison
val_tuk <-
  pmod %>%

```

```

aov() %>%
  TukeyHSD(which= "Treatment") %>%
  broom::tidy() %>%
  rename(p = adj.p.value) %>%
  mutate(estimate = paste0(signif(estimate, 2)),
         p = case_when(
           p >= 0.05 ~ '> 0.05',
           between(p, 0.001, 0.009) ~ '< 0.01',
           p < 0.001 ~ '< 0.001',
           TRUE ~ paste0('= ', round(p, 2)) ))

# create table of Tau results with tidy dplyr pipe using purrr
kt <-
  ha %>%
  mutate(Date = format(Date, "%j"),
         Date = as.numeric(Date)) %>%
  split(.$Treatment) %>%
  map(~ kendallTrendTest(Hoppers ~ Date, data = .)) %>%
  map_dfr(~ as.data.frame(t(as.matrix(c(round(.$p.value, 3),
                                         .$estimate[1:2],
                                         .$interval$limits)))) %>%

  mutate(season = c("Fall", "Spring", 'Unburned')) %>%
  rename( p = z) %>%
  select(season, tau, p, slope, LCL, UCL) %>%
  mutate(tau = as.character(signif(tau, 2)),
         p = case_when(
           p >= 0.05 ~ '> 0.05',
           between(p, 0.001, 0.009) ~ '< 0.01',
           p < 0.001 ~ '< 0.001',
           TRUE ~ paste0('= ', round(p, 2)) ))

```