

Appendix B. Worked Examples for Plant Cover Data Analysis

Written by Kathryn M. Irvine, US Geological Survey, kirvine@usgs.gov

February 1, 2019

This appendix is written in an .Rnw file format that can be opened and viewed within R studio (R Core Team 2017). The R code requires loading the following R packages from CRAN: ggplot2 (Wickham 2009), plyr (Wickham 2011), xtable (Dahl 2016), tidyr (Wickham and Henry 2017), ordinal (Christensen 2015), betareg (Cribari-Neto and Zeileis 2010), VGAM (Yee 2017), rjags (Plummer 2016), mosaic (Pruim et al. 2017). The rjags package requires a working installation of JAGS (Just Another Gibbs Sampler) (Plummer 2015a). The R code that is not mirrored in the compiled .pdf contains the code to manipulate the data and the code to re-create all the displayed tables and figures. This document is fully reproducible, if the user can negotiate the R software.

The datasets analyzed herein are available for download at <https://doi.org/10.5066/P9OHMLL1> along with this file.

1 Sago pondweed (*Stuckenia pectinata*) Example: Line-intercept or visually estimated cover in an areal plot

Field methods that employ a line intercept or visually estimating cover within an areal plot result in a response variable that is a continuous proportion. In the main text, we presented a field study interested in whether there was evidence of a difference in mean plant cover after the water was removed within a wetland unit. Seventy 1-m x 1-m plots were randomly placed throughout the wetland unit located at Bowdoin National Wildlife Refuge (Montana, USA) in summer of 2015 and the following summer (2016) after the water was removed, 62 plots were surveyed. These data were collected by Bowdoin Refuge Wildlife Biologist Jessica Larson as part of a larger inventory and monitoring project within the US Fish and Wildlife Service.

For analysis, one could use a beta regression model (betareg function in betareg R package) or logit-transformed response and apply linear regression options (lm function in stats R package). For some situations, the assumptions of the linear regression model are violated and they should be checked. The betareg model objects and the lm model objects both allow for the plot function to create diagnostic plots to evaluate model assumptions. For brevity, we display only one set of diagnostic plots for a beta regression model (code for other fitted model objects is included in .Rnw file for reference). Data may display non-constant variance, outliers, etc and just using a transformation is insufficient. In our example, there were a large number of plots with zero recorded cover. One option is to transform the zero values to trace ($= 0.005$) and 1 to slightly less than 1 ($= 0.995$) and then fit the models (beta regression versus linear model). A better option as discussed in the text is to use a zero-one augmented beta model, which we also demonstrate. Basically a three-part model, the zero

and/or one values are modeled separately using binary logistic regression and the continuous cover values that are not 0 or 1 are modeled using beta regression.

1.1 Data Visualization for Sago Pondweed Example

The histograms show the empirical distribution on the raw proportion scale and the logit-transformed scale (Fig. B1). In this example, we replaced zero values with a trace ($=0.005$) and one values with 0.995. The histogram is the same, if we had used 0 or 1 values. The data appear L-shaped in 2015 and then U-shaped in distribution in 2016 (Fig. B1 top row). This shape remains even after the logit-transformation (Fig. B1 bottom row). These data are typical in that they have a lot of zero values recorded for one species (Table B1).

Total Sample Size	Number of Zeros	Number of Ones
132	87	10

Table B1: Summary of Observations from Plot based visual estimates of Sago Pondweed

1.2 Methods for Analyzing Sago Pondweed

As described in the main text, one option for dealing with the 0 and 1 values is to transform them to be slightly less than one or more than zero. This approach assumes that the data are consistent with a common beta distribution. We fit five models to the dataset: three variations on the beta model and two linear model approaches.

A beta regression assuming a common spatial aggregation δ or precision parameter (ϕ) (object named: `mod.beta1`). Notice that $\delta = \frac{1}{1+\phi}$ and $\phi = \frac{(1-\delta)}{\delta}$. A beta regression assuming each year had a different ϕ parameter (object named: `mod.beta2`).

Another option for modeling the data is to use a zero-one augmented beta model. Currently, the `betareg` package does not implement this model directly. Therefore, we follow the theoretical results shown in Ospina and Ferrari (2010) that suggest a three-part model can be fit to the data. Basically, we use logistic regression with response an indicator variable for whether or not the plot had zero recorded cover, another logistic regression with response an indicator for whether or not the plot had 100% recorded percent cover, and then the beta regression is used to model the continuous percent cover observations ranging from greater than 0 and less than 1.

Other options based on assuming that the residuals are normally distributed is to use a linear model with a logit-transformed response (object named: `mod.lmlogit`) or a linear model with response untransformed proportions (object named: `mod.lmraw`).

```
#this is using the betareg function within the package betareg in R

mod.beta1<-betareg(prop.STPE15.t~yrunit,data=dat.BDW, link = c("logit"), link.phi = NULL,
                    type = c("ML"))
```

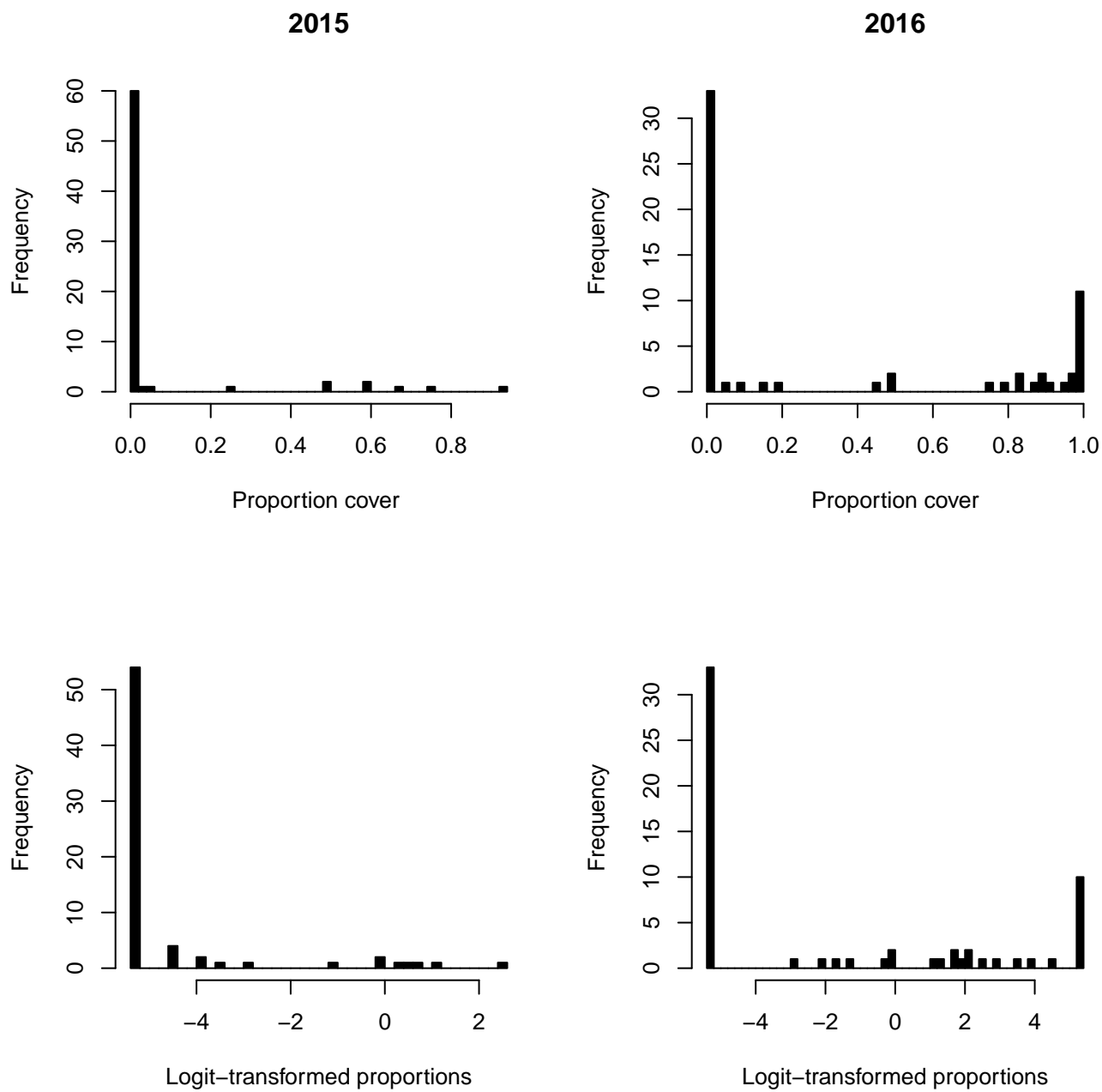


Figure B1: Comparison of non-Logit versus logit transformed Sago Pondweed percent cover with 0 and 1 values replaced with 0.005 and 0.995, respectively.

```

# allowing for the phi parameter to among years
mod.beta2<-betareg(prop.STPE15.t~yrunit|yrunit,data=dat.BDW,link = c("logit"), link.phi = NULL,
                  type = c("ML"))

# For comparison this applies a logit-transformation to the empirical proportions and
# then uses a standard linear regression model.

mod.lmlogit<-lm(Sago.logit~yrunit,data=dat.BDW)

#this is code for applying a naive linear model applied to the raw proportions as done
#in the simulation study

mod.lmraw<-lm(prop.STPE15.t~yrunit,data=dat.BDW)

```

```

#Demonstration on how to extract AIC from beta regression models

aic.betamodel1<- AIC(mod.beta1)
aic.betamodel2<- AIC(mod.beta2)

```

1.3 Interpreting Results for Sago Pondweed

In order to choose between the beta regression model with a common ϕ (Table B2) versus different ϕ (Table B3), I used AIC but a likelihood ratio or wald test could be used. Using AIC, the model with varying ϕ values had a lower AIC (-464 compared to -432) and therefore more support. We consider our simple two year or group case to explain how to interpret the output from beta regression.

The model we fit assumes

$$\text{logit}(\mu_j) = \beta_0 + \beta_1 \text{Ind}_{grp2},$$

where Ind_{grp2} is an indicator for group 2 and j denotes the group membership so $j = 1$ or $j = 2$. We have $\text{logit}(\mu_2) - \text{logit}(\mu_1) = \beta_1$, which is equivalent to

$$\log\left(\frac{\mu_2}{1 - \mu_2}\right) - \log\left(\frac{\mu_1}{1 - \mu_1}\right) = \beta_1.$$

The $\frac{\mu_j}{1 - \mu_j}$ is interpreted as the odds of proportion cover in group j . Therefore,

$$\log\left(\frac{\mu_2}{1 - \mu_2} / \frac{\mu_1}{1 - \mu_1}\right) = \beta_1$$

is the log- odds ratio of cover in group 2 compared to group 1,

$$\left(\frac{\mu_2}{1 - \mu_2} / \frac{\mu_1}{1 - \mu_1}\right) = \exp(\beta_1).$$

Then $\exp(\beta_1)$ is the factor increase/decrease in odds of proportion cover for group 2 compared to group 1, where $\exp(\beta_1) > 1$ is an increase and $\exp(\beta_1) < 1$ is a decrease, and $\exp(\beta_1) \approx 1$ means essentially no change.

Interpreting the beta regression coefficient labeled as *yrunit2016* in Table B3 suggests that the odds of proportion sago pondweed cover increased by a factor of $5.9 = \exp(1.776)$ in 2016 from 2015. Alternatively, the logit-linear model suggested that odds of sago cover increased by a factor of $17 = \exp(2.866)$ in 2016 from 2015 (Table B4). The linear model on raw proportions suggested that sago cover increased by 0.293 in 2016 from 2015 (Table B5). These estimates all suggest an increase, however, the magnitude of the effect is quite different.

The residual diagnostic plots suggest that modeling the zeros separate from the non-zeros fit the observed data the best (Fig. B3). The results from fitting the zero-one augmented beta model suggest that a common ϕ is better supported with $AIC = -16.8155$ (Table B6) versus allowing ϕ to vary among years $AIC = -14.9304$ (Table B7). Based on the zero-one augmented beta model, the odds of sago proportion cover increase by a factor of 4 ($\exp(1.44)$) in 2016 compared to 2015 (Table B6). The odds of zero cover decreased by a factor of 0.33 ($\exp(-1.08)$) in 2016 (Table B8). Notice the model fit to just the ones versus non-ones did not fit the data well, likely due to the fact there were zero plots in 2015 observed with 100% sago pondweed and only 10 in 2016 (Table B9). To address this issue, we fit a beta model to the observations greater than 0 and changed the one values to 0.995. The interpretation was essentially the same, odds of proportion sago increased by a factor of 7 in 2016 compared to 2015 (model output not shown).

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-1.251	0.165	-7.602	0.000
yrunit2016	0.716	0.221	3.246	0.001
(phi)	0.796	0.087	9.163	0.000

Table B2: Results from using betareg package in R by transforming the 0 and 1's. These are on the logit-scale for μ with a common ϕ parameter

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-2.186	0.200	-10.940	0.000
yrunit2016	1.776	0.266	6.677	0.000
phi (Intercept) 2015	1.009	0.211	4.785	0.000
phi (yrunit2016) 2016	-1.580	0.252	-6.266	0.000

Table B3: Results from using betareg package in R by transforming the 0 and 1's. Allowing for ϕ to vary by unit. These are on the logit-scale for μ and log-scale for ϕ

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-4.491	0.391	-11.485	0.000
yrunit2016	2.866	0.571	5.023	0.000

Table B4: Results from using lm package in R with logit-transformed percent cover for sago pondweed both years

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.075	0.040	1.855	0.066
yrunit2016	0.293	0.059	4.992	0.000

Table B5: Results from using lm package in R with raw percent cover for sago pondweed both years

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.876	0.305	-2.876	0.004
yrunit2016	1.444	0.425	3.395	0.001
(phi)	1.530	0.317	4.823	0.000

Table B6: Results from using betareg package in R for continuous portion of model. These are on the logit-scale for μ

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.841	0.327	-2.574	0.010
yrunit2016	1.429	0.429	3.330	0.001
phi (Intercept) 2015	0.348	0.312	1.113	0.266
phi (yrunit2016) 2016	0.141	0.418	0.338	0.735

Table B7: Results from using betareg package in R for for continuous portion of model. Varying ϕ . These are on the logit-scale for μ . These are on the logit-scale for μ and log-scale for ϕ

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.2164	0.2846	4.27	0.0000
yrunit2016	-1.0872	0.3818	-2.85	0.0044

Table B8: Results for fitting logistic regression model for probability of zero sago pondweed after water was removed.

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-20.5661	2119.1845	-0.01	0.9923
yrunit2016	18.9174	2119.1846	0.01	0.9929

Table B9: Results for fitting logistic regression model for probability of complete cover of sago pondweed after water was removed.

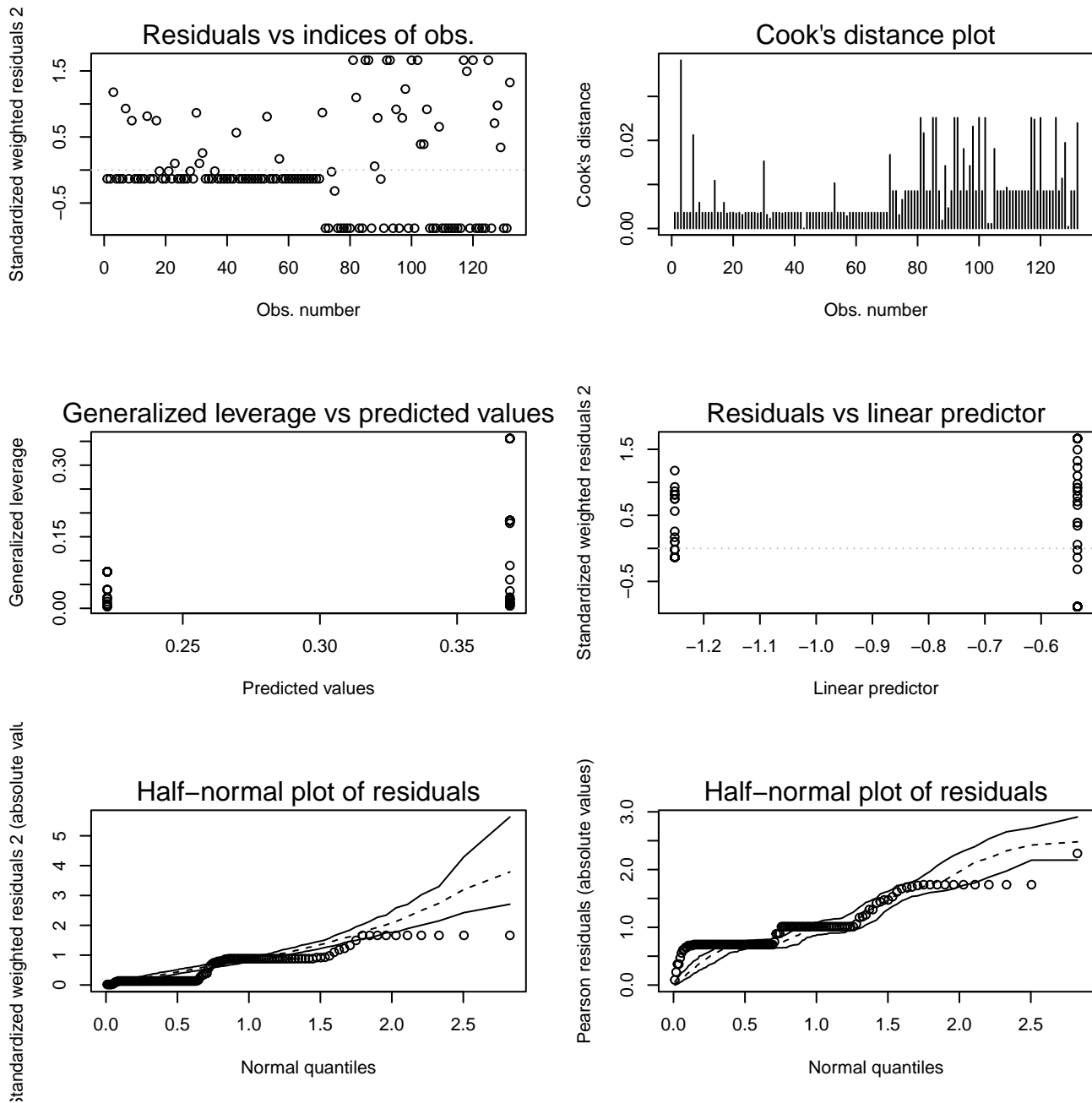


Figure B2: Example diagnostic plots for beta regression (R code presented in .Rnw version)

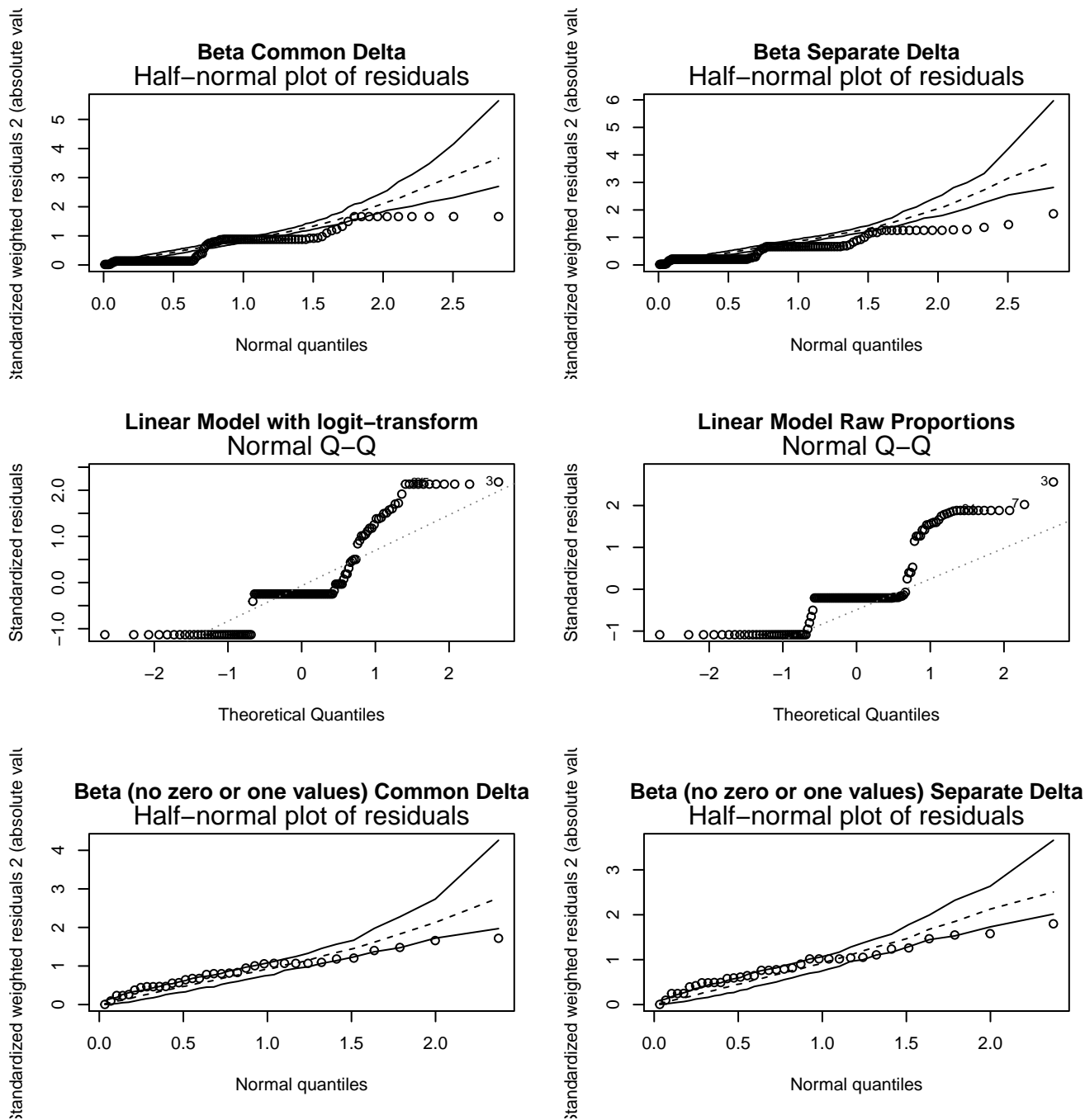


Figure B3: Comparison of default qqplots for beta and linear models. Beta models use weighted residuals 2 recommended in Espinheira et al. 2008 and linear model are the response residuals.

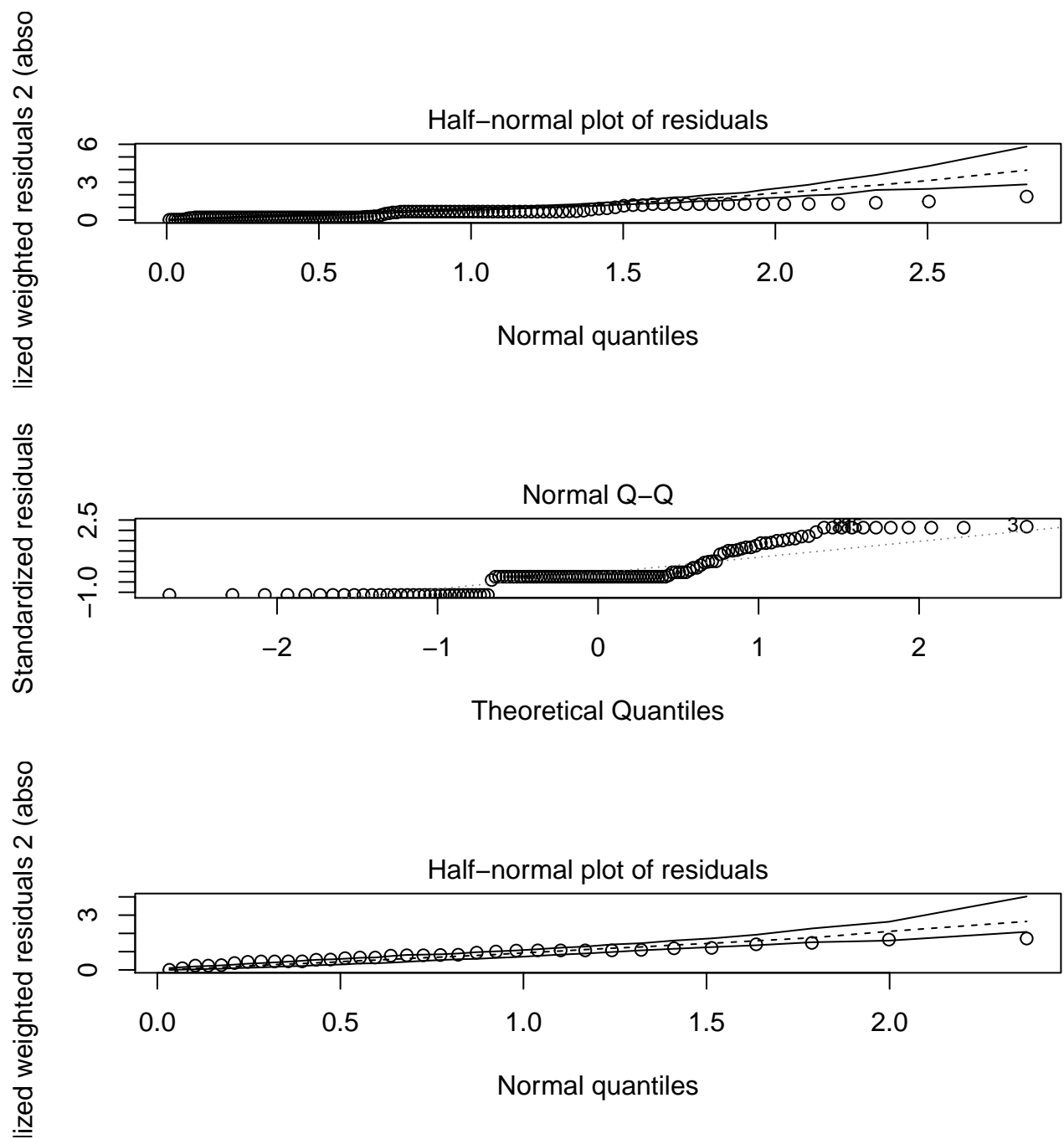


Figure B4: Comparison of default qqplots for beta and linear models. Beta models use weighted residuals 2 recommended in Espinheira et al. 2008 and linear model are the response residuals.

2 Pin-point or point-intercept datasets

Assuming recorded data is number of times a pin hit the species of interest with a known number of pins as arises with pin-point or point intercept data collection.

I simulated data assuming one site was repeatedly measured for 7 years and there were 100 plots or lines within each year. For analysis, one could use a beta-binomial with the VGAM package or R-INLA, we demonstrate using the VGAM package(object named: mod.betabinom). The VGAM package uses 'rho' as the spatial aggregation parameter we refer to as δ . A logistic regression for binomial counts is an option, but a check for over-dispersion should be done as empirical plant cover data are subject to spatial aggregation that may manifest in overdispersed binomial counts (object named: mod.glm). Alternatively, a logit-transformed response can be used in linear regression options (object named: mod.lm). For some situations, the assumptions for the linear regression models are violated and they should be checked as we show below. Data may display non-constant variance, outliers, etc and just using a transformation is insufficient.

delta	beta_0	beta_1	phi
0.33	0.50	0.20	2.00

Table B10: True Beta Parameter Values Used to Simulate Data

#This is fitting a beta-binomial model using the VGAM package and the vglm function.

```
mod.betabinom <- vglm(cbind(Y.betabin, c(rep(30,700)-Y.betabin)) ~ Year, betabinomial(zero = "rho"),
  data = dat.betatrend, trace = TRUE)
```

	Coefficients	Lower 95% CI	Upper 95% CI
logit(delta)	-0.66	-0.77	-0.55
beta_0	0.51	0.42	0.60
beta_1	0.17	0.12	0.21

Table B11: Estimated Values and Wald Confidence Intervals for Beta-Binomial with VGAM pkg

*# for comparison for pin-point data, a logistic regression for binomial counts is fit using
glm function in Base R.*

```
mod.glm<-glm(cbind(Y.betabin, c(rep(30,700)-Y.betabin)) ~ Year,
  data=dat.betatrend,family=binomial(link=logit))
```

Another approach is to use logit-transformation and then apply a standard linear regression model.

```
mod.lm<-lm(logit.bin~Year,data=dat.betatrend)
```

```
#built-in diagnostic plots for glm models
```

```
par(mfrow=c(2,2),pty='m')
```

```
plot(mod.glm)
```

```
#built-in diagnostic plots for lm models
```

```
par(mfrow=c(2,2),pty='m')
```

```
plot(mod.lm)
```

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.5194	0.0145	35.72	0.0000
Year	0.1907	0.0074	25.94	0.0000

Table B12: Results for fitting binomial logistic regression model for trend

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.0500	0.0947	11.09	0.0000
Year	0.3915	0.0473	8.27	0.0000

Table B13: Results for fitting linear model for trend with logit-transformed proportions

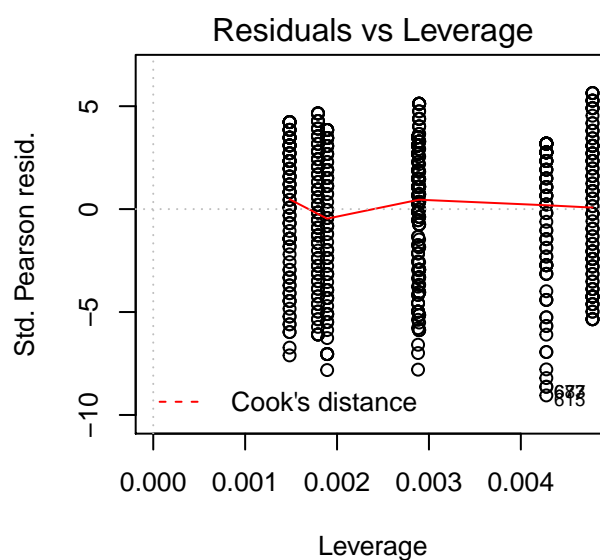
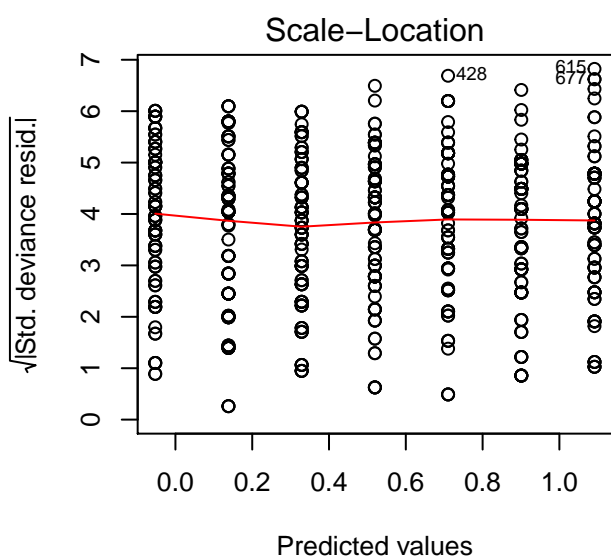
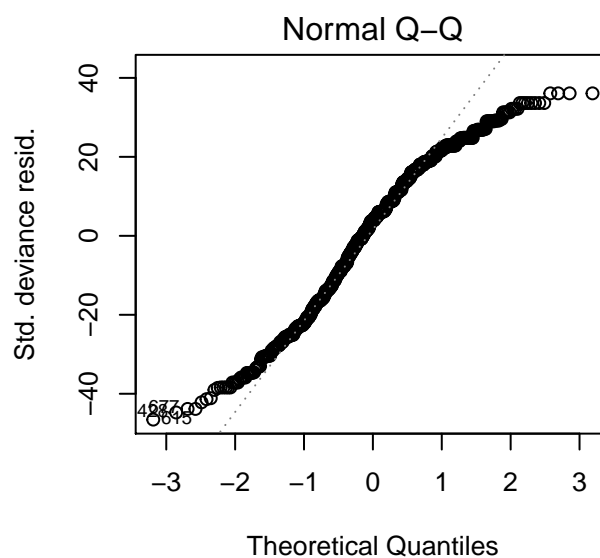
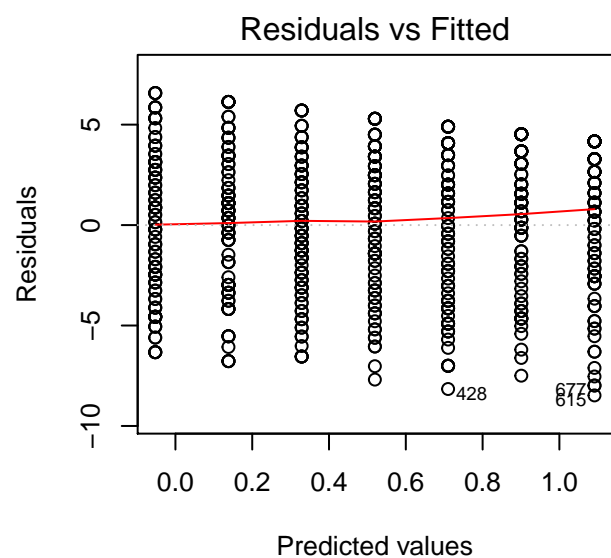


Figure B5: Diagnostic plot for logistic regression for binomial counts

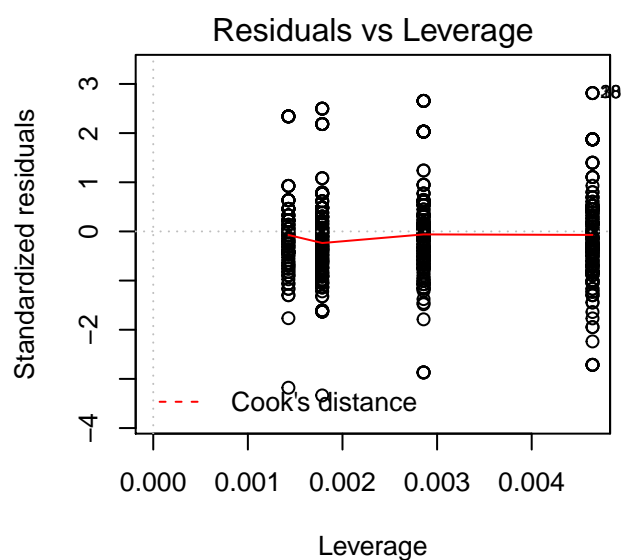
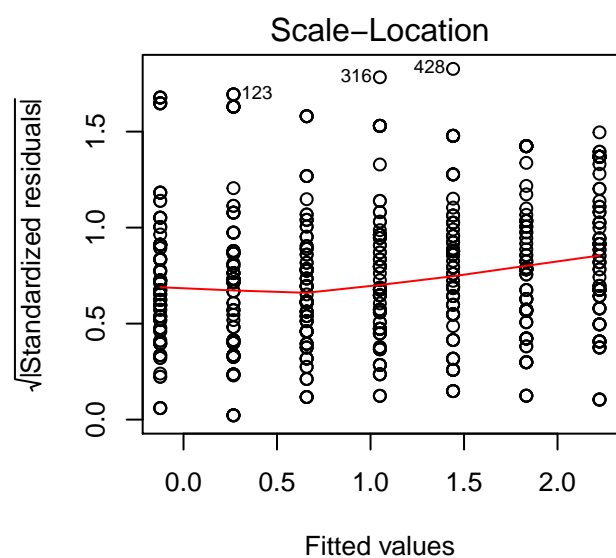
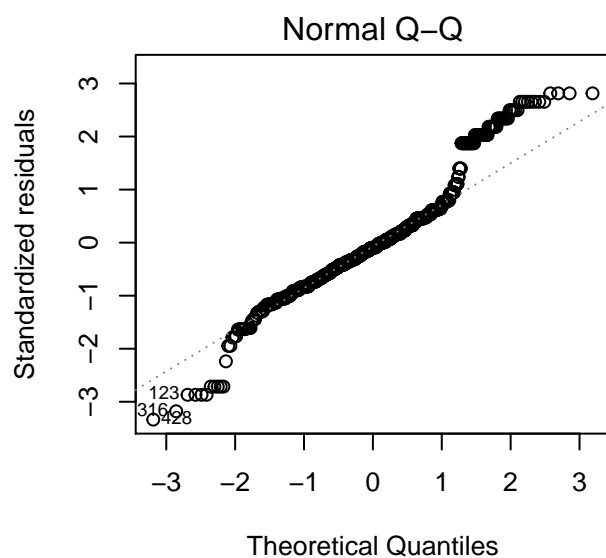
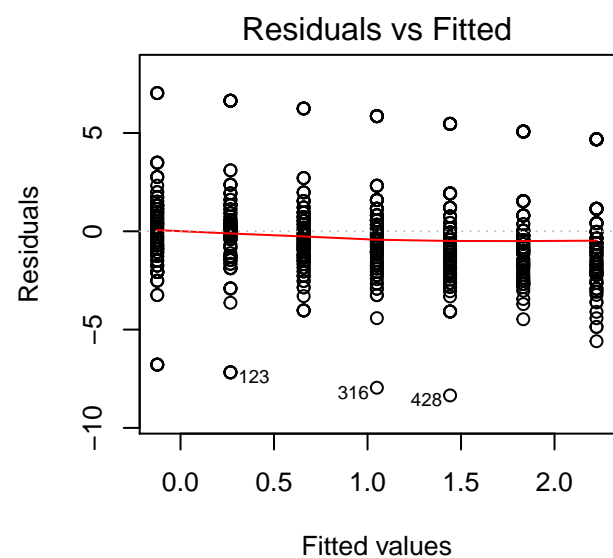


Figure B6: Diagnostic plot for logit-transformation of empirical proportions and applying linear regression

3 Ordinal data trend analysis: cheatgrass (*Bromus tectorum*) example

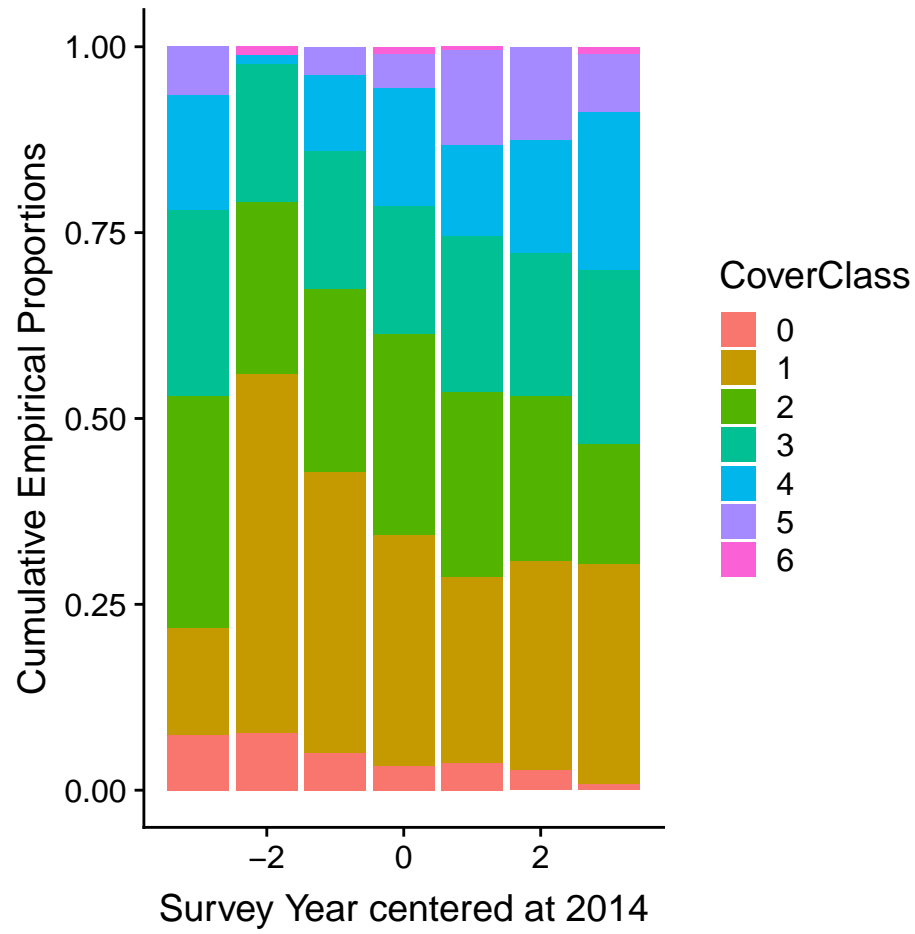
For plant abundance that is measured by visually estimating the top cover within an areal plot and then placing in a cover class based on a pre-defined scheme. A commonly used cover class scheme is based on a modified Daubenmire scale:

Cover Class	Range	Midpoint
0	0	0
1	> 0 – 5%	2.5 %
2	> 5 – 25%	15 %
3	> 25 – 50%	37.5 %
4	> 50 – 75%	62.5 %
5	> 75 – 95%	85 %
6	> 95%	97.5 %

The following example analysis is based on data recorded as ordinal cover classes of cheatgrass (*Bromus tectorum*), a nonnative invasive grass. The data were collected every year from 2011 until 2017 at the Clarno unit of the John Day Fossil Beds National Monument. The areal plots were placed following a random selection of locations based on the GRTS algorithm (Yeo et al. 2009). There was a fire documented in 2011 and interest is in assessing whether there is evidence of cheatgrass distribution or abundance increasing in the monument post-fire.

3.1 Exploratory Data Analysis

A bar plot is a useful graphics to explore empirical proportions for cover classes versus either a factor or continuous covariate. A barplot reflects the true ordinal categorical nature of cover classes. In this example the explanatory variable of interest was the year of survey, but a discretized version of a continuous variable would work as well as (see Irvine *et al.*, 2016, Figure 1).



	Count
0	57
1	403
2	337
3	287
4	196
5	104
6	6

Table B14: category values

3.2 Proportional Odds Model

The proportional odds model (POM) can be fit using the ordinal package in R or JAGS, WinBUGS, Stan for Bayesian inferences. I include code and results for using ordinal and JAGS.

Using the `clm` function parameterizes the model as

$$\text{logit}(P(Y_i \leq c)) = \theta_c - \beta_1 \text{Year}_i$$

for plot $i = 1, \dots, n$ and $c = 1, \dots, C - 1$. So the interpretation becomes in terms of the odds of moving into a higher category across years. However, with a quadratic term this means basically a \cup or \cap shape to the temporal trajectory. In my opinion, graphics are critical in this case.

Following our advice in Irvine et al. 2016, we use a diagnostic plot to see if we need to separately model the zero class from the non-zero classes. This plot displays the different cumulative logits. Notice if there are very few observations in some categories the uncertainty will be very large. This plot can help determine if the proportional odds assumption is violated if the coefficients are not similar across the different cumulative logits (Fig. B6).

```
#using a cumulative logit model for the cover class data
```

```
mod.POM <- clm(CoverClass ~ Year, data = dat.ordinal)
```

	Coefficients	Std. Errors	z value	Pr(> z)
0 1	-3.146	0.135	-23.253	0.000
1 2	-0.695	0.057	-12.155	0.000
2 3	0.308	0.055	5.648	0.000
3 4	1.286	0.065	19.715	0.000
4 5	2.483	0.100	24.859	0.000
5 6	5.474	0.409	13.374	0.000
Year	0.089	0.024	3.697	0.000

Table B15: POM for Annual Trend in Ordinal Cover Classes. Year is the estimated trend parameter and $c|c + 1$ parameters are the corresponding threshold or cut-point parameters.

```
# Ordinal data with observed categories of {0,...,6} recoded to {1,...,7}.
```

```
#save the POM.jags, OZAB.jags, and datafile to the same working directory space
```

```
#If this model doesn't compile double-check that the data take on the following values
```

```
#table(dat.JAGS$yo)
```

```
# Table output from call above
```

```
#  1  2  3  4  5  6  7
```

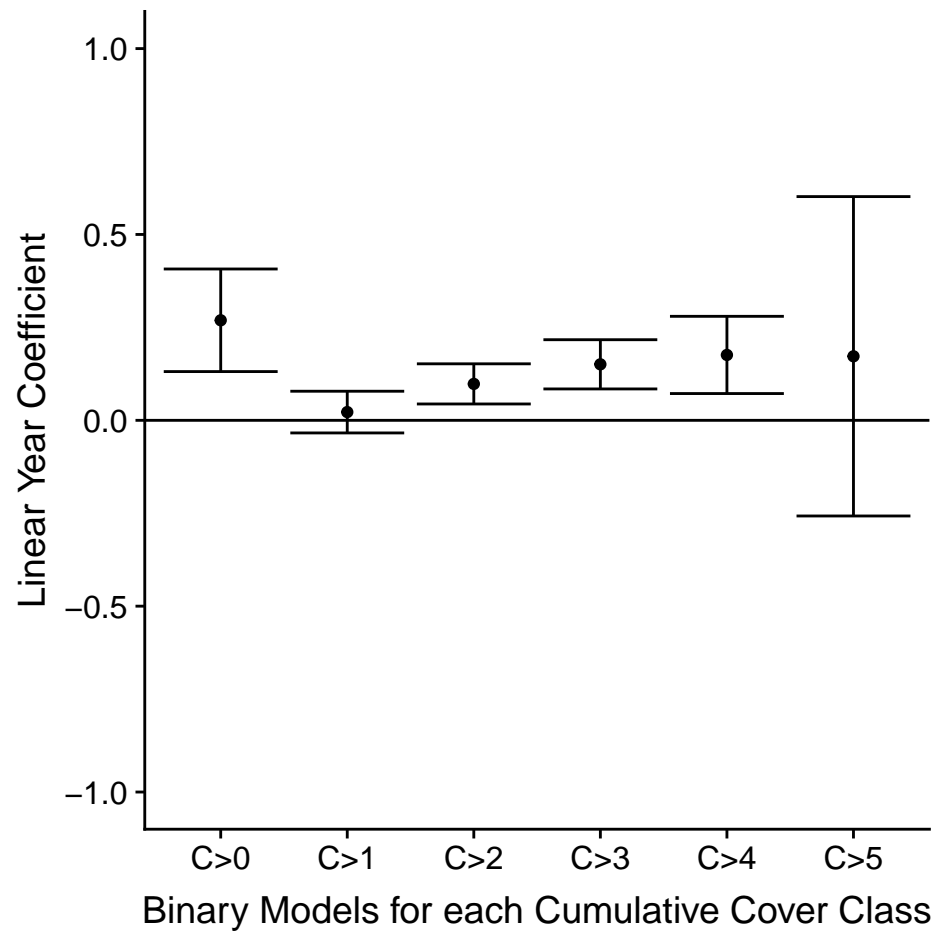



Figure B7: Diagnostic Plot to assess proportional odds assumption $\Pr(C > j)$. This plot suggests the need to separate the zero class from the non-zero classes.

```

# 57 403 337 287 196 104 6
#

#####

##### 1. POM model using latent variable construction
#####

#create X-matrix for predictors of interest
X.beta<-model.matrix(~ (-1)+Year, dat.ordinal)

#create dataset for JAGS
dat.JAGS<-list("yo"=as.numeric(dat.ordinal$Y.ord)+1,"X"=X.beta, "N"=dim(dat.ordinal)[1],
               "P"=dim(X.beta)[2], "J"=length(unique(dat.ordinal$Y.ord)))

#jags model call: Make sure you save the POM.jags file so you can call it next

jm.PO<- jags.model('POM.jags' , dat.JAGS , n.chains = 4 ,
                  inits = function(dat = dat.JAGS , ...){
                    zz <- sort(runif(dat$J-1))
                    z0 <- c(0 , zz)
                    y.l <- (z0 + diff(c(z0,1))/2)[dat$yo]
                    list(
                      beta = rnorm(dat$P , 0 , 5),
                      zeta0.l = zz ,
                      y.latent = logit(y.l)
                    )
                  },
                  quiet=TRUE)

#If this model doesn't compile double-check that the data take on the following values

#table(dat.JAGS$yo)

```

```

# Table output from call above
#   1   2   3   4   5   6   7
# 57 403 337 287 196 104   6
#

update(jm.PO , 5000)

# 5000 iterations with thinning by 4; returns an mcmc.list object

#sage.PO.jags <- coda.samples(jm.PO , c('beta','zeta.est','llike','Dev') , 5000 , 4)

sage.PO.jags <- coda.samples(jm.PO , c('beta','zeta.est') , 5000 , 4)

#take a look MCMC chains and posteriors for parameters set in previous coda.samples call
# WARNING: notice these look pretty bad as we would expect due to the zeros being included !
# included for illustration only

```

	Point est.	Upper C.I.
beta	1.00	1.00
zeta.est[1]	1.05	1.14
zeta.est[2]	1.37	1.97
zeta.est[3]	1.60	2.36
zeta.est[4]	1.15	1.39
zeta.est[5]	1.08	1.22
zeta.est[6]	1.00	1.00

Table B16: Gelman Diagnostic

	x
beta	4741.14
zeta.est[1]	71.69
zeta.est[2]	31.48
zeta.est[3]	28.76
zeta.est[4]	25.88
zeta.est[5]	55.44
zeta.est[6]	638.39

Table B17: Effective Sample Size for each Parameter

3.3 Ordinal Hurdle Models: Dealing with zero cover class in cover class datasets

These models allow for the zeros to be modeled with different predictors/covariates than the non-zeros or abundance cover classes (see Irvine *et al.* 2016 for more details). I specify a linear trend for the zero vs non-zeros and similarly a linear trend

	2.5%	25%	50%	75%	97.5%
beta	0.04	0.07	0.09	0.10	0.14
zeta.est[1]	-3.40	-3.24	-3.15	-3.07	-2.93
zeta.est[2]	-0.80	-0.74	-0.71	-0.68	-0.64
zeta.est[3]	0.21	0.27	0.30	0.33	0.38
zeta.est[4]	1.18	1.24	1.28	1.32	1.40
zeta.est[5]	2.28	2.38	2.43	2.49	2.64
zeta.est[6]	4.67	5.09	5.33	5.59	6.17

Table B18: Posterior Summaries for each Parameter fitting Bayesian Proportional Odds Model. Beta is the estimated linear trend parameter and zeta.est parameters are the cut-point or threshold parameters (θ_C in model equation.)

for the cumulative logits of the non-zeros. This can be done using the ordinal package and removing the zero class. And then a separate logistic regression for estimating probability of non-zero classes or probability of detecting cheatgrass. I only include the MLE version, but could be done with Bayesian estimation.

```
##### Cheatgrass Ordinal Hurdle Model
#####

dat.sub.ordinal<- subset(dat.ordinal, subset=Y.ord!=0)
mod.nozero <- clm(CoverClass~ Year, data = dat.sub.ordinal)

#zeros only
mod.PA <- glm(Detection~ Year, family=binomial, data =dat.ordinal)
```

	Coefficients	Std. Errors	z value	Pr(> z)
1 2	-0.83	0.06	-13.89	0.00
2 3	0.23	0.06	4.15	0.00
3 4	1.22	0.07	18.70	0.00
4 5	2.43	0.10	24.28	0.00
5 6	5.42	0.41	13.24	0.00
Year	0.06	0.02	2.47	0.01

Table B19: POM for Annual Trend in Ordinal abundance only model

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	3.2331	0.1467	22.04	0.0000
Year	0.2691	0.0704	3.82	0.0001

Table B20: Annual Trend for Detection with Logistic regression for zeros vs. non-zeros

3.4 Ordinal Zero-Augmented Beta Model: Another Option for Modeling Zero Cover Class in Daubenmire scale datasets

These models specify the latent distribution as beta and was introduced in Irvine *et al.* (2016). I fit a OZAB (ordinal zero-augmented beta model) which specifies the zeros separately. This method is useful when there is a meaningful zero class in the measurement scheme. In this two-part model, the parameter estimates can be interpreted as follows. For a given

predictor k included in the beta model, the interpretation of $\exp(\beta_k)$ is in terms of a multiplicative change in the odds of the proportion of a plot covered by a species with a unit-change in X_k , given all other predictors are held at a fixed value. For a given predictor l that is related to the presence or absence of a species, the interpretation of $\exp(\gamma_l)$ is the effect of a one-unit change in W_l on the odds of a zero class, given all other predictors are held at a fixed value.

Importantly, for a more intuitive interpretation, the inverse logit transformation of $\mathbf{X}_i\beta$ provides inferences regarding the mean of the latent abundance process or the proportion of a plot covered (e.g., percent areal cover/100). Similarly, the probability of absence (zero class) can be explored using the inverse logit transformation of $\mathbf{W}_i\gamma$. Larger values for the precision parameter, ϕ , produce less variability in percent cover values and is related to the spatial aggregation parameter δ by $\delta = \frac{1}{1+\phi}$.

```
#####

#####

##### 2. Fitting OZAB (ordinal zero-augmented beta model)

#####

#####

#nozeros refers to the subset of data with cover class > 0
#separate estimation for abundance and detection-process

# creating dataset for JAGS

X.beta<-model.matrix(~ (-1)+Year,dat.sub.ordinal)
X.bern<-model.matrix(~Year,dat.ordinal)

OZAB.dat<-
  list("y.beta"=as.numeric(dat.sub.ordinal$Y.ord),"X.beta"=X.beta,
       "N.beta"=dim(dat.sub.ordinal)[1],"P.beta"=dim(X.beta)[2],
       "y.bern"=dat.ordinal$Detection,"N.bern"=length(dat.ordinal$Detection),"X.bern"=X.bern,
       "K"=dim(X.bern)[2])

#compiling takes awhile...

#The values for y.beta should be 1 to 6

#set a list of initial values to assist with error messages, I used the MLE estimates
#as starting values sometimes, but could generate from random starting values
```

```

#
initsList<-
  list(list("beta"=c(.06),"phi"=1,"alpha"=c(2,.26),
    "y.latent"=c(.025 , .15 , .375 , .625 , .85 , .975)[OZAB.dat$y.beta]),
    list("beta"=c(0),"phi"=3,"alpha"=c(0,0),
    "y.latent"=c(.025 , .15 , .375 , .625 , .85 , .975)[OZAB.dat$y.beta]),
    list("beta"=c(-.05),"phi"=4,"alpha"=c(-1,-.25),
    "y.latent"=c(.025 , .15 , .375 , .625 , .85 , .975)[OZAB.dat$y.beta]))

jm.sage.new <- jags.model('OZAB.jags' , OZAB.dat , n.chains=3 , n.adapt = 5000 ,
  quiet=TRUE ,inits = initsList)

update(jm.sage.new , 5000)

# 5000 iterations with thinning by 4; returns an mcmc.list object

#OZAB.sage.jags <- coda.samples(jm.sage , c('alpha','beta','phi','dev.Beta','dev.O') , 5000 , 4)
OZAB.sage.jagsnew <- coda.samples(jm.sage.new , c('alpha','beta','phi') , 5000 , 4)

```

	Point est.	Upper C.I.
alpha[1]	1.00	1.00
alpha[2]	1.00	1.00
beta	1.00	1.00
phi	1.00	1.00

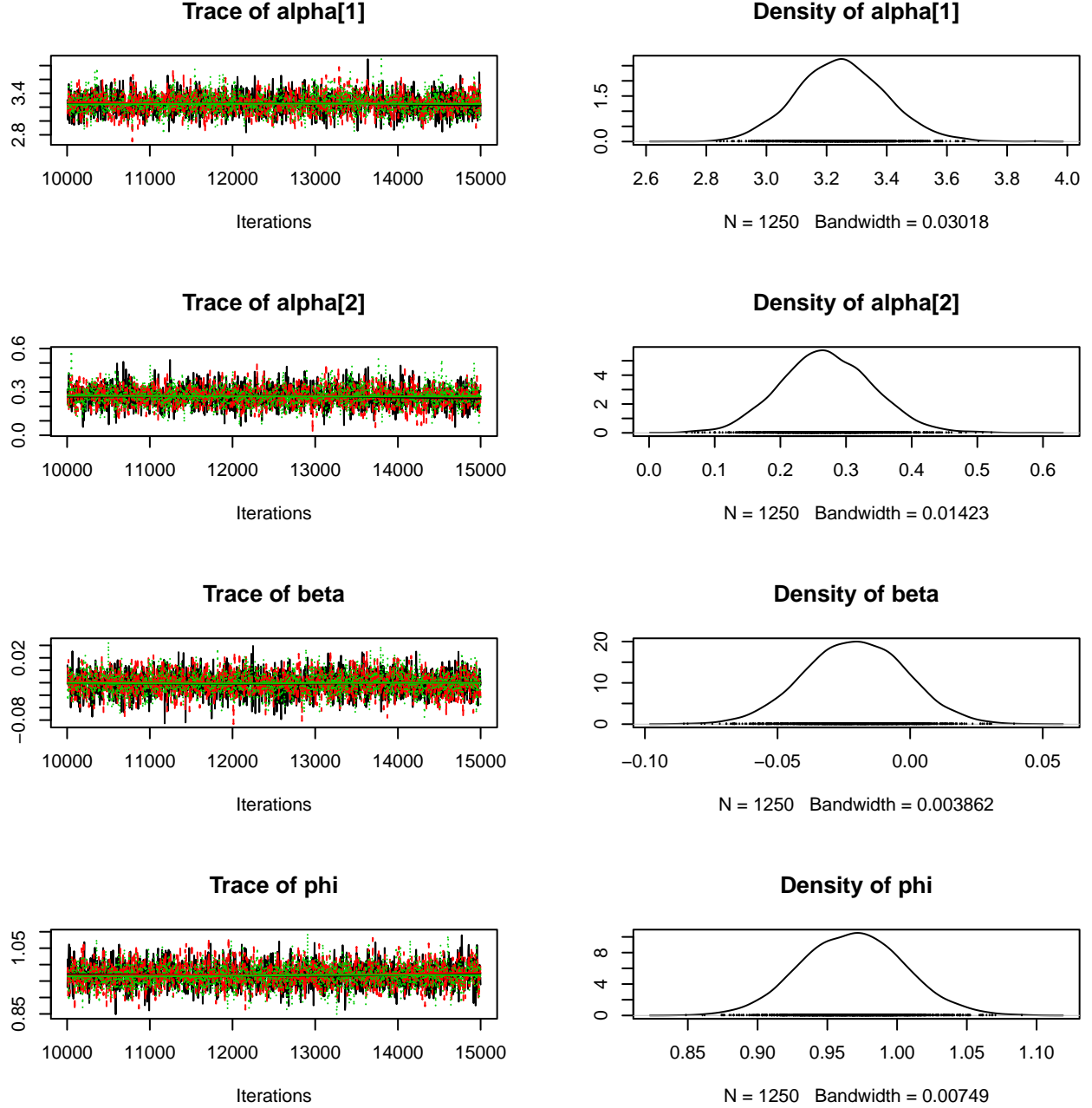
Table B21: Gelman Diagnostic for OZAB

	x
alpha[1]	3360.32
alpha[2]	3260.44
beta	3689.03
phi	3118.02

Table B22: Effective sample size for each Parameter OZAB

	2.5%	25%	50%	75%	97.5%
alpha[1]	2.97	3.15	3.25	3.35	3.55
alpha[2]	0.14	0.22	0.27	0.32	0.41
beta	-0.06	-0.03	-0.02	-0.01	0.02
phi	0.90	0.94	0.97	0.99	1.04

Table B23: Posterior Summaries for each Parameter in OZAB. Alpha parameters correspond to estimating probability of detection. Beta is linear coefficient on year term on logit-scale for percent cover and phi is dispersion parameter that is related to spatial aggregation parameter.



3.5 Ordinal Beta Model

I created an artificial “trace” category to incorporate the zeros into the beta model. These results were basically the same as separating out the zeros for the abundance portion of the OZAB model.

```

#just create a small threshold for 0's so add to the beta portion of the model
# I used 1% for the zero class

X.beta<-model.matrix(~ (-1)+Year,dat.ordinal)

OBM.dat<-
  list("y.beta"=as.numeric(dat.ordinal$Y.ord)+1,"X.beta"=X.beta,
       "N.beta"=dim(dat.ordinal)[1],"P.beta"=dim(X.beta)[2])

#Here y.beta needs to be coded as 1 to 7 to merge the zeros into a trace category

initsList<-
  list(list("beta"=c(-0.049 ),"phi"=1,
            "y.latent"=c(.01, .025 , .15 , .375 , .625 , .85 , .975)[OBM.dat$y.beta]),
        list("beta"=c(0),"phi"=3,
            "y.latent"=c(.01,.025 , .15 , .375 , .625 , .85 , .975)[OBM.dat$y.beta]),
        list("beta"=c(.05),"phi"=4,
            "y.latent"=c(.01,.025 , .15 , .375 , .625 , .85 , .975)[OBM.dat$y.beta]))

jm.sage.new1 <- jags.model('OrdBeta.jags' , OBM.dat , n.chains=3 , n.adapt = 5000 ,
                          quiet=TRUE ,inits = initsList)

update(jm.sage.new1 , 5000)

OHM.sage.jagsnew1 <- coda.samples(jm.sage.new1 , c('beta','phi') , 5000 , 4)

```

	Point est.	Upper C.I.
beta	1.00	1.00
phi	1.00	1.00

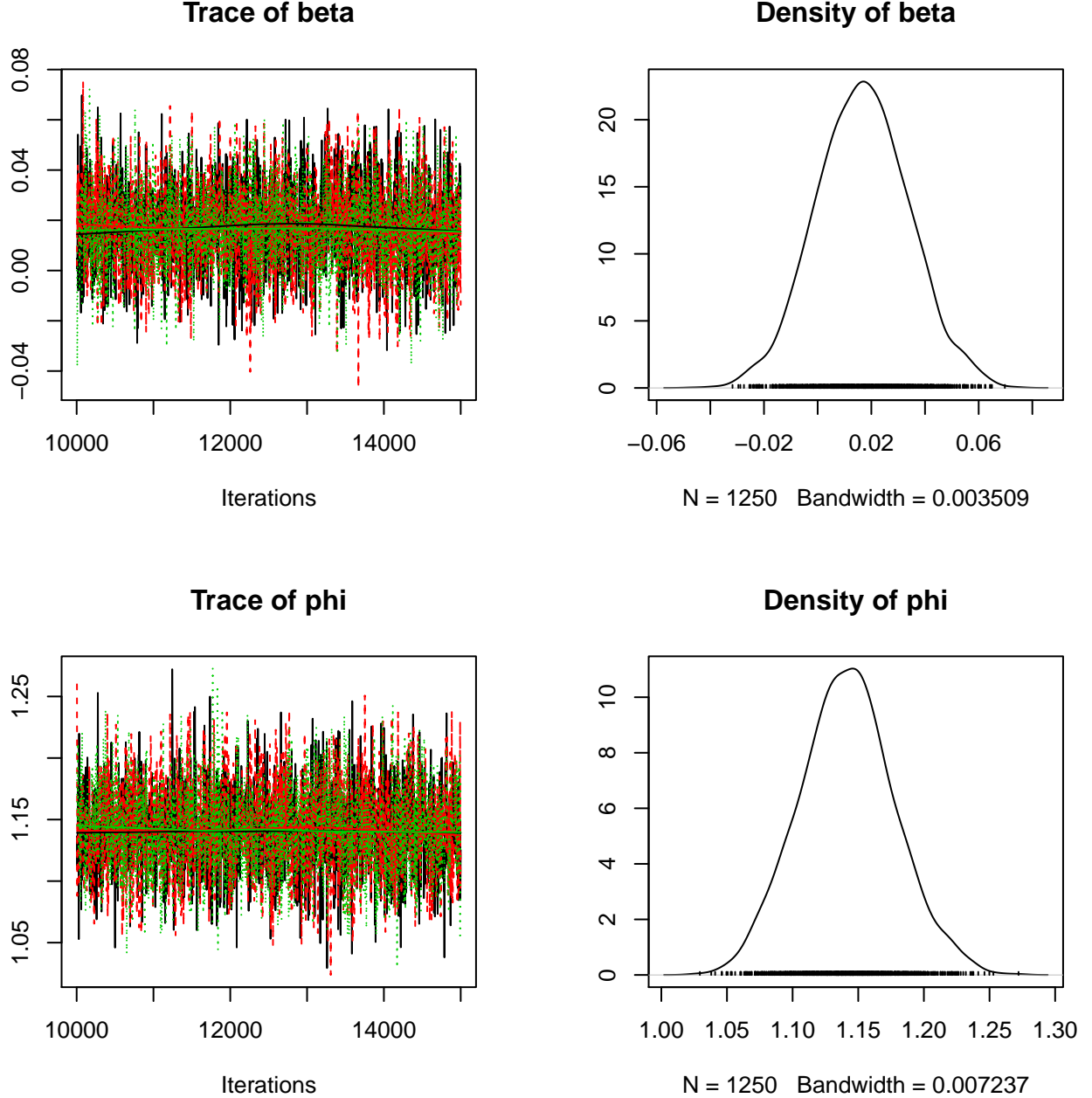
Table B24: Gelman Diagnostic for Ordinal Beta Model

	x
beta	3750.00
phi	3803.91

Table B25: Effective sample size for each Parameter Ordinal Beta Model

	2.5%	25%	50%	75%	97.5%
beta	-0.02	0.01	0.02	0.03	0.05
phi	1.07	1.12	1.14	1.16	1.22

Table B26: Posterior Summaries for each Parameter in Ordinal Beta Model. Beta is linear coefficient on year term on logit-scale for percent cover and phi is dispersion parameter that is related to spatial aggregation parameter.



3.6 Analyzing Ordinal data as midpoint values using naive linear model

This approach is included as a comparison. It is not recommended for many reasons that are related to the bounded nature of ordinal data. For example, the model could predict outside of the range possible values for the response variable. As mentioned in the text, percent cover and cover class data are inherently heteroskedastic (nonconstant variance) in that there is more variation for moderate values and less towards the boundaries of 0 or 1.

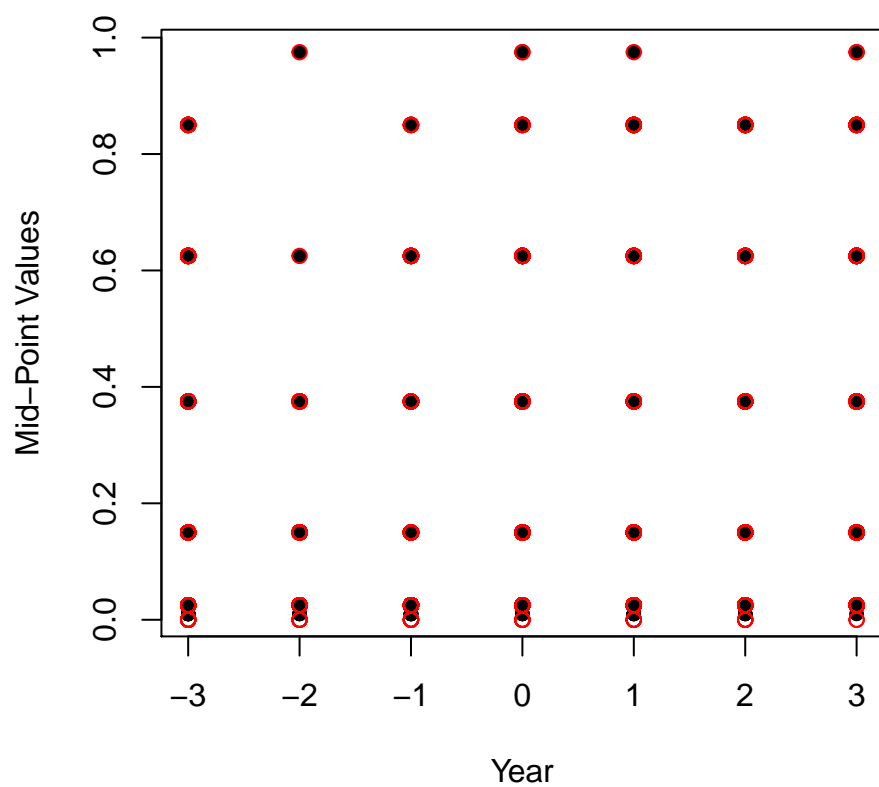


Figure B8: Plot of Midpoints versus year centered at 2014 (black=trace value for zeros). Included as a comparison to show how little information is conveyed versus the stacked barplot in main text Fig. 5.

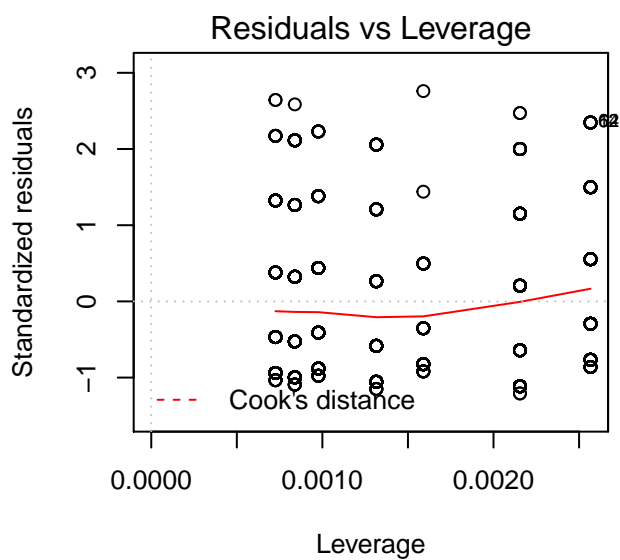
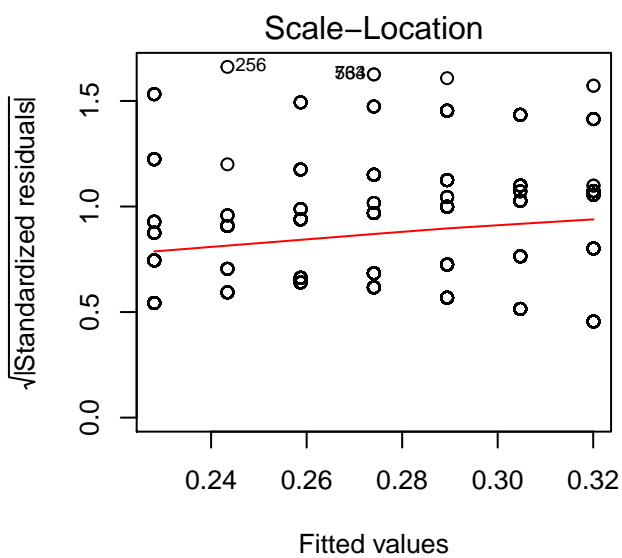
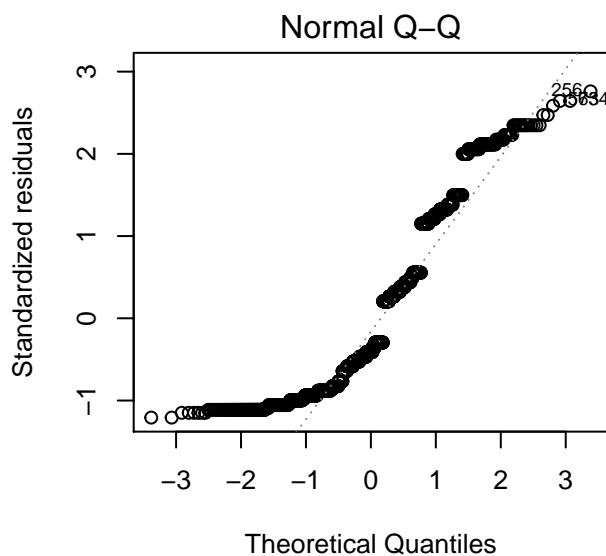
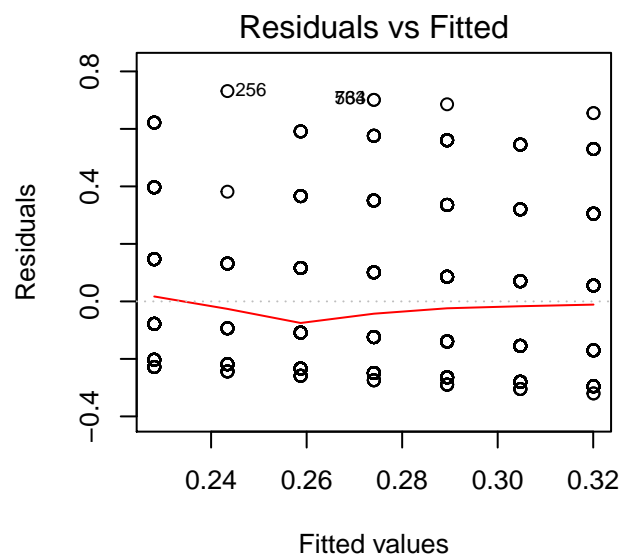


Figure B9: Diagnostic plots for applying linear regression to midpoints

	Estimate	Std. Error	t value	$\Pr(> t)$
(Intercept)	0.2741	0.0071	38.35	0.0000
Year	0.0153	0.0036	4.29	0.0000

Table B27: Model summary based on fitting linear model to midpoint values

Notice that the normal qq-plot in Figure B8 shows outliers (labeled with the corresponding observation number), left-skewness in the lower tail of the distribution, and striping reflects the inherent categorical nature of the data.

Literature Cited

- Christensen, R. H. B. (2015). *ordinal—Regression Models for Ordinal Data*. R package version 2015.6-28.
<http://www.cran.r-project.org/package=ordinal/>.
- Cribari-Neto, F. and A. Zeileis (2010). Beta Regression in R. *Journal of Statistical Software* 34:1–24. URL:
<http://www.jstatsoft.org/v34/i02/>.
- Dahl, D. B. (2016). *xtable: Export Tables to LaTeX or HTML*. R package version 1.8-2. URL:
<https://CRAN.R-project.org/package=xtable>.
- Espinheira, P. L., S. L. Ferrari, and F. Cribari-Neto (2008). On beta regression residuals. *Journal of Applied Statistics* 35:407–419.
- Irvine, K. M., T. J. Rodhouse, and I. N. Keren (2016). Extending ordinal regression with a latent zero-augmented beta distribution. *Journal of Agricultural, Biological, and Environmental Statistics* 21:619–640.
- Plummer, M. (2015a). JAGS Version 4.0.0 user manual: URL: <https://sourceforge.net/projects/mcmc-jags/files/Manuals/4.x>.
- Plummer, M. (2016). *rjags: Bayesian Graphical Models using MCMC*. R package version 4-6. URL:
<https://CRAN.R-project.org/package=rjags>.
- Pruim, R., D. T. Kaplan, and N. J. Horton (2017). The mosaic Package: Helping Students to 'Think with Data' Using R. *The R Journal* 9:77–102. URL: <https://journal.r-project.org/archive/2017/RJ-2017-024/index.html>.
- R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: <https://www.R-project.org/>.
- Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN: 978-0-387-98140-6. URL:
<http://ggplot2.org>.
- Wickham, H. (2011). The Split-Apply-Combine Strategy for Data Analysis. *Journal of Statistical Software* 40:1–29. URL:
<http://www.jstatsoft.org/v40/i01/>.
- Wickham, H. and L. Henry (2017). *tidyr: Easily Tidy Data with 'spread()' and 'gather()' Functions*. R package version 0.7.2. URL:
<https://CRAN.R-project.org/package=tidyr>.
- Yee, T. W. (2017). *VGAM: Vector Generalized Linear and Additive Models*, R package version 1.0-4. URL:
<https://CRAN.R-project.org/package=VGAM>.
- Yeo, J., T. J. Rodhouse, G. H. Dicus, K. M. Irvine, and L. K. Garrett (2009). Sagebrush steppe vegetation monitoring protocol. Upper Columbia Basin Network. Version 1.0. Natural Resource Report NPS/UCBN/NRR-2009/142. National Park Service, Fort Collins, CO, USA: