ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Applied Biostatistics - MATH-493

Assignment 3: Generalized Linear Models in R, Horseshoe Crab Dataset

Authors:

Wong Wen Yan,

Professor:

Darlene Goldstein

June 11, 2018

Contents

1 Introduction

The dataset chosen for this study is the Horseshoe Crab dataset. The goal of this study is to predict the number of spines present in a horseshoe crab, given 5 independent variables:

1. `y`: whether the female crab has a satellite (`1` = yes, `0` = no), binary
2. `weight`: weight in grams, real valued and positive
3. `color`: color of horseshoe crab, can take on four possible values (1, 2, 3, 4), categorical
4. `width`: width of the female crab in centimeters, real valued and positive
5. `sat`: number of satellites (`y` = 0 if and only if `sat` = 0), positive integer

The dependent variable `spine` is discrete and can only take on positive values. In particular, `spine` only has three possible values: 1, 2 and 3. Since the dependent variable is a count variable, a linear regression model, which has the form

$$Y = \beta^T X + \epsilon\,, \quad \epsilon \sim \mathcal{N}(0,\, \sigma^2)$$
$$\beta, X \in \mathbb{R}^n$$

will not be able to represent the discrete and non-negative nature of `spine` because ϵ is a continuous random variable which can take values from $-\infty$ to $+\infty$. To model a count variable, we need to consider generalized linear models with discrete and non-negative distribution functions (eg. Poisson distribution, Negative Binomial distribution). In other words, the model must fulfill

$$f_Y(y) = \begin{cases} \exp(a(y)b(\theta) + c(\theta) + d(y)), & \text{if } y \in \mathbb{Z}^+ \\ 0, & \text{otherwise} \end{cases} \tag{1}$$
$$\eta(E(Y|X)) = \beta^T X$$

where η is the link function, and $\beta, X \in \mathbb{R}^n$.

In this study, three instances of such generalized linear models will be used to predict `spine`, namely: ***Poisson regression model***, ***Quasi Poisson regression model***, and ***Negative Binomial regression model***. The best models of each type will be compared against each other using multiple metrics and criteria. Finally, the best model will be selected.

2 Initial Data Analysis

Before fitting models to the data, exploratory data analysis was carried to gain better understanding of the structure of the dataset, using common visualization tools presented below. These visualizations also help to reveal the relative importance of the independent variables in predicting `spine`.

2.1 Overview of dataset

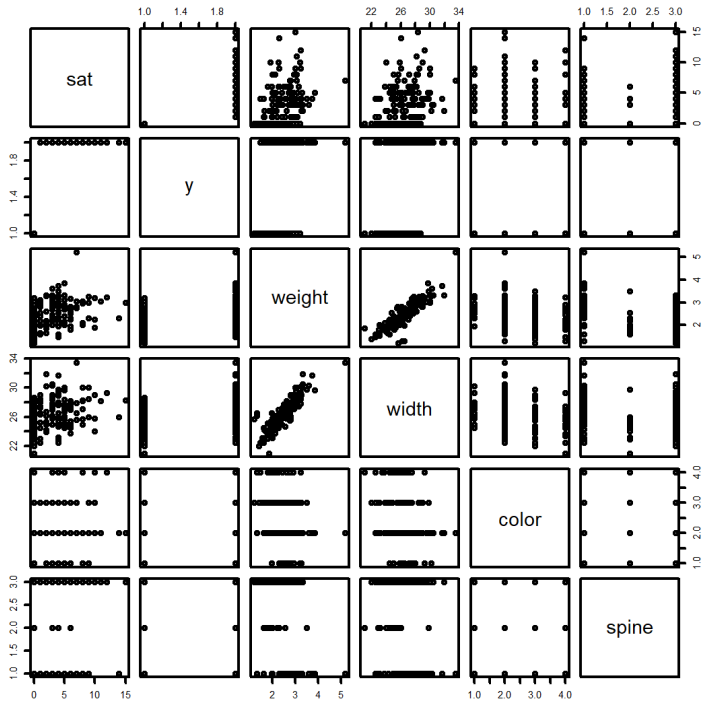
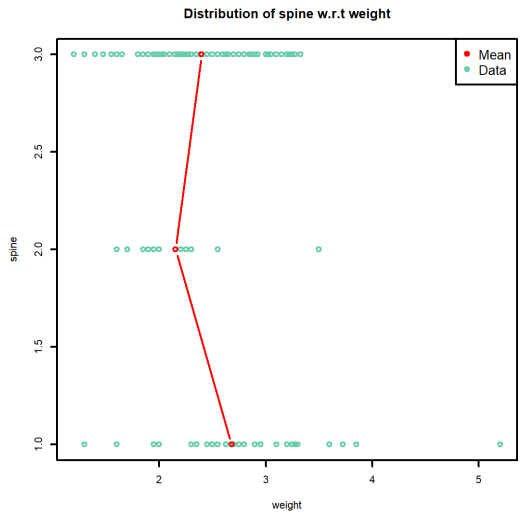


Figure 1: Scatter matrix of the Horseshoe Crab dataset. Key observations: (a) `spine`, `color`, `y` and `sat` are indeed discrete valued. (b) No noticeable strong correlation between `spine` and any one of the independent variables. (c) Strong correlation between `weight` and `width`, this may cause multicollinearity issues in modeling if both variables are included.

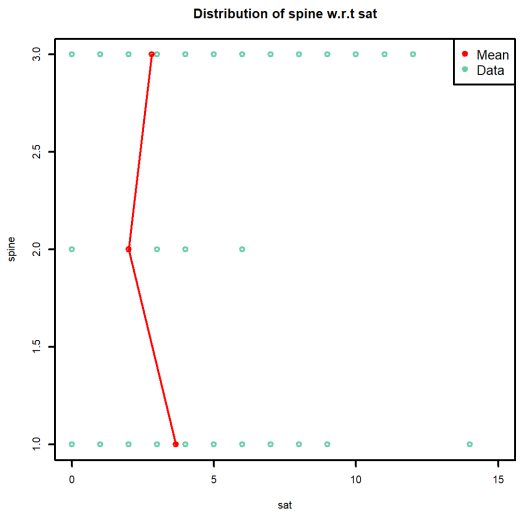
2.2 Importance of numeric predictors

	sat	weight	width	spine
sat	1.00	0.37	0.34	-0.09
weight	0.37	1.00	0.89	-0.17
width	0.34	0.89	1.00	-0.12
spine	-0.09	-0.17	-0.12	1.00

Table 1: Correlation Matrix between numeric predictors and `spine`. Since `y` and `color` are categorical, their numerical correlation with `spine` is not meaningful for interpretation. Key observations: (a) No significant linear dependence between `spine` and any of the `spine` and any of the numeric predictors. Numeric predictors might be less useful. (b) Strong positive correlation between `weight` and `width` (0.89), this will very likely cause multicollinearity issues.



(a) Distribution of `spine` w.r.t `weight`



(b) Distribution of `spine` w.r.t `sat`

Figure 2: Red points indicate the mean of predictor when `spine` is fixed at a certain value. If we split the dataset into three classes based on the value of `spine`, it is clear that the intra-class variances of `weight` and `sat` are very large. On top of that, the differences in inter-class means of `weight` and `sat` are relatively small. We can expect `weight` and `sat` themselves be less useful in predicting `spine`. Besides, the distribution of `spine` w.r.t `width` is not displayed here, but it is expected to behave in a similar fashion as `weight` since they are highly correlated.

2.3 Importance of categorical predictors

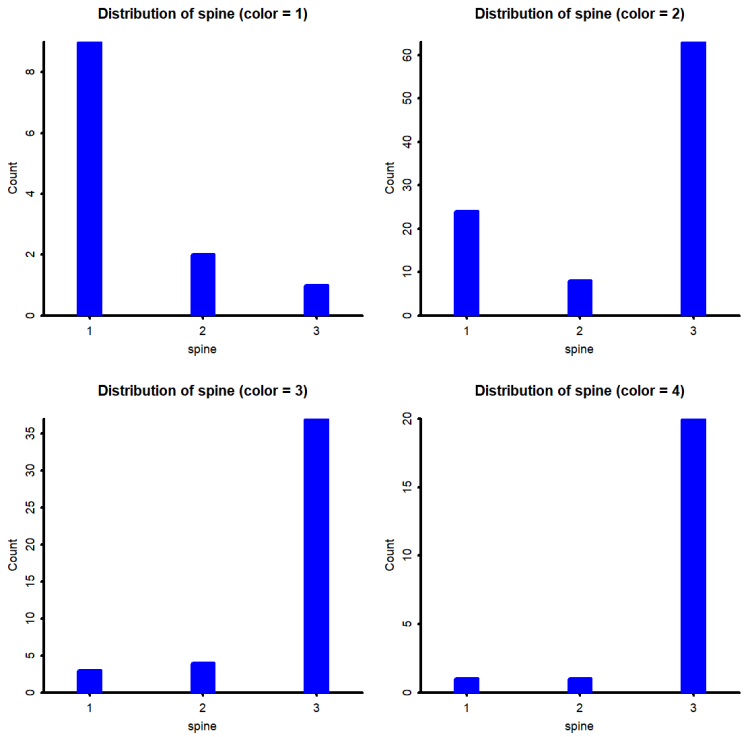


Figure 3: Histogram to illustrate the distribution of `spine` when `color = 1`, `color = 2`, `color = 3` and `color = 4`.The distribution of `spine` when `color = 1` is very different compared to when `color = 2, 3, 4`. Most of the time, when `color = 1`, `spine` takes the value `1`. When `color` is 2, 3 or 4, `spine` is very likely going to be 3. `spine = 2` occurs much less frequently compared to the other two cases. It appears that `color` is likely going an important predictor for modeling `spine`.

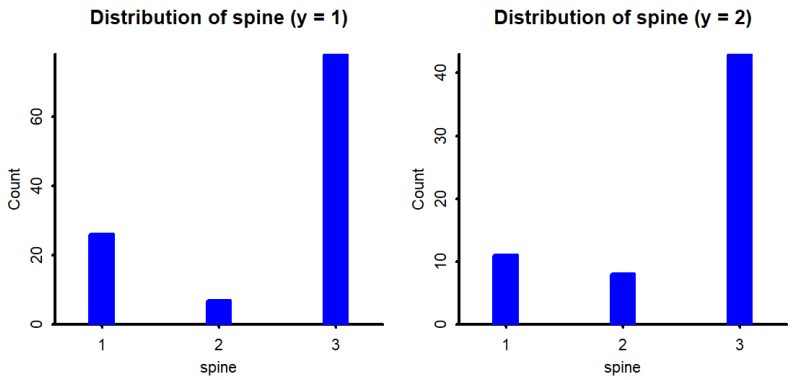


Figure 4: Histogram to illustrate the distribution of `spine` when `y = 0` and `y = 1`. These two histograms are very similar in terms of shape. This indicates that the predictor `y` itself is probably not very useful for predicting `spine`.

2.4 Distribution of `spine` with respect to 2 variables

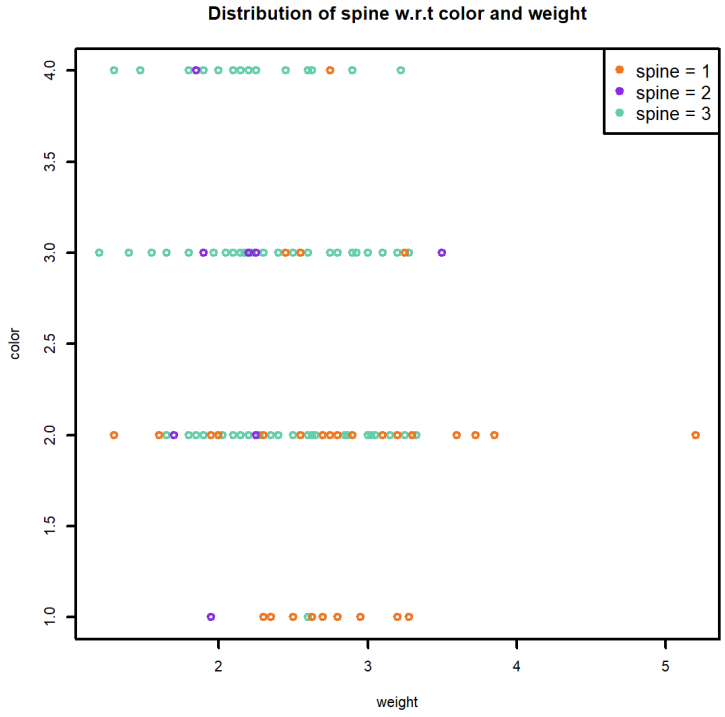


Figure 5: Distribution of `spine` w.r.t `color` and `weight`. Orange points (`spine = 1`) occur more frequently on the bottom-right corner, while green points (`spine = 3`) are more prevalent on the top half. The fact that we can visually detect a structure might indicate that `color:weight` is a useful interaction be included in modeling.

3 Poisson regression

The density function of Poisson distribution can be written as

$$P(Y=y) = \begin{cases} \frac{\mu^y e^{-\mu}}{y!} & \text{if } y \in \mathbb{Z}^+ \\ 0 & \text{otherwise} \end{cases}$$
$$= \begin{cases} \exp(y\log(\mu) - \mu - \log(y!)) & \text{if } y \in \mathbb{Z}^+ \\ 0 & \text{otherwise} \end{cases}$$

which fulfills equation (1). In a Poisson regression model, μ is simply $E(Y|X)$, the mean of response variable Y given X, and it is associated with the linear predictor $\beta^T X$ via the log link function

$$\log(E(Y|X) = \beta^T X$$

Since $E(Y|X) = \exp(\beta^T X)$, regardless of the values of X, $E(Y|X)$ will always be positive. Additionally, since the distribution of Y given X is Poisson, it follows that

$$Var(Y|X) = E(Y|X) = \exp(\beta^T X)$$

The single parameter μ in a Poisson distribution defines its entire distribution. This means that, in a Poisson regression model, the linear predictor $\beta^T X$ determines both variance and mean of Y. Although this is an elegant formulation, for many datasets, $Var(Y|X)$ may be much larger or smaller than $E(Y|X)$. Fitting a Poisson regression model to such data may lead to *overdispersion* or *underdispersion*.

In this section, the goal is to obtain the best possible Poisson regression model to serve as a benchmark before moving on to more complex models.

3.1 Experimentation

3.1.1 Baseline model with all main effects

The first Poisson model that was attempted includes all main effects available, `spine = sat + y + weight + width + color`. The summary of coefficients (Figure 6) show that estimates of the coefficients of all predictors are not significant at the 0.05 level, except for `color`.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.00954	0.979	-0.00975	0.992
sat	-0.00723	0.0222	-0.326	0.744
y1	0.0918	0.148	0.622	0.534
weight	-0.132	0.18	-0.732	0.464
width	0.0224	0.0493	0.455	0.649
color2	0.582	0.259	2.24	0.0248
color3	0.717	0.269	2.67	0.00763
color4	0.766	0.285	2.69	0.00716

Figure 6: Coefficients of Poisson regression model `spine = sat + y + weight + width + color` fitted to dataset. Only indicator variables of `color` are significant at the 0.05 level.

3.1.2 Single predictor model

The model `spine = sat + y + weight + width + color` may be a reference that can be used for comparison later on. Since it is much more difficult to understand a model with large

number of predictors, the strategy adopted here is to start with the simplest possible model, and then build upon it. To find such a model, all possible models with single predictor was fitted to the data.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.15	0.211	5.44	5.45e-08
weight	-0.0976	0.0854	-1.14	0.253

(a) `spine = weight`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.923	0.0801	11.5	9.89e-31
y1	-0.0191	0.1	-0.191	0.849

(b) `spine = y`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.42	0.607	2.34	0.0195
width	-0.0193	0.0231	-0.837	0.403

(c) `spine = width`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.938	0.0654	14.3	1.09e-46
sat	-0.00965	0.0156	-0.618	0.537

(d) `spine = sat`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.288	0.25	1.15	0.25
color2	0.592	0.259	2.29	0.022
color3	0.732	0.266	2.75	0.00589
color4	0.764	0.28	2.73	0.00632

(e) `spine = color`

Figure 7: Coefficients from Poisson regression models with single predictor. `spine = color` has the best fit here. For models in (a), (b), (c) and (d) the coefficients of predictor variable are all not significant at the 0.05 level.

Based on Figure 7, it is clear that `color` is the most important variable for predicting `spine`. To further justify this, the *likelihood ratio test* can be used to compare all of the single predictor models against a reduced (or null) model, `spine = 1`. The null model simply predicts `spine` using its sample average.

Formally, a likelihood-ratio test is a hypothesis test with

$$H_0 : \text{reduced model is true}, \quad H_1 : \text{alternative model is true}$$

$$\Delta G^2 = -2(\log(\mathcal{L}_0(\beta_0; X, y)) - \log(\mathcal{L}_1(\beta_1; X, y))$$

where ΔG^2 is the test statistic, $\mathcal{L}_0(\beta_0; X, y)$ and $\mathcal{L}_1(\beta_1; X, y)$ are the likelihoods of null and alternative models. ΔG^2 follows chi-squared distribution with k degrees of freedom, where k is the number of additional coefficients that are present in alternative model but absent in reduced model. Explicit expression of ΔG^2 for likelihood ratio tests between two Poisson models can be derived easily using the expression for likelihood function of a Poisson model :

$$\mathcal{L}(\beta; X, y) = \prod_{i=1}^{n} \frac{\exp(\beta^T X_i)^{y_i} e^{-\exp(\beta^T X_i)}}{y_i!}$$

The likelihood-ratio test can only be used if the reduced and alternative models are nested, otherwise ΔG^2 will no longer follow chi-squared distribution. In our case, `spine = 1` is a valid reduced model for all other Poisson models discussed so far. Additionally, this also allows us to use the AIC, which is defined as

$$\text{AIC} = 2p - \log(\mathcal{L}(\beta; X, y)), \quad p = \text{number of estimated parameters}$$

to compare `spine = 1` with every other Poisson model. Table 2 illustrates the results of likelihood ratio tests and AIC values.

Model		Log-likelihood		AIC		LR test: P-value
Reduced	Proposed	Reduced	Proposed	Reduced	Proposed	
spine = 1	spine = sat + y + weight + width + color	-265.69	-259.91	533.38	535.82	0.12
spine = 1	spine = color	-265.69	-260.41	533.38	528.82	0.01
spine = 1	spine = weight	-265.69	-265.03	533.38	534.05	0.25
spine = 1	spine = width	-265.69	-265.34	533.38	534.67	0.40
spine = 1	spine = sat	-265.69	-265.50	533.38	534.99	0.53
spine = 1	spine = y	-265.69	-265.67	533.38	535.34	0.85

Table 2: Likelihood-ratio tests and AIC values of each all models with single predictors. Key observations: (a) `spine = color` is the only single predictor model that passes the likelihood ratio test against null model at 0.05 significance level. (b) `spine = color` is the only model with lower AIC than null model. (c) Despite having 4 more predictors, `spine = sat + y + weight + width + color` is only slightly better than `spine = color` in terms of log likelihood, -259.91 - (-260.41) = 0.5

3.1.3 Improving upon `spine = color`

Results from Table 2 confirms that `spine = color` is the best single predictor model. This is in fact consistent with observations made during exploratory analysis (Figure 3). The next step is to consider models of the form `spine = color + ?`, where `?` can be any one of `sat`, `y`, `width` or `weight`. Figure 8 presents the fitted coefficients for these models.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.293	0.258	1.14	0.256
color2	0.591	0.259	2.28	0.0224
color3	0.73	0.267	2.73	0.00637
color4	0.762	0.282	2.7	0.00687
sat	-0.00134	0.0157	-0.085	0.932

(a) `spine = color + sat`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.413	0.339	1.22	0.223
color2	0.588	0.259	2.27	0.0232
color3	0.716	0.268	2.68	0.00742
color4	0.743	0.283	2.63	0.00864
weight	-0.0478	0.0872	-0.548	0.584

(b) `spine = color + weight`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.425	0.689	0.617	0.537
color2	0.591	0.259	2.28	0.0224
color3	0.726	0.267	2.71	0.00665
color4	0.756	0.283	2.67	0.00753
width	-0.00509	0.0238	-0.214	0.831

(c) `spine = color + width`

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.258	0.262	0.985	0.324
color2	0.593	0.259	2.29	0.0218
color3	0.738	0.266	2.77	0.00558
color4	0.781	0.284	2.75	0.00588
y1	0.0388	0.105	0.369	0.712

(d) `spine = color + y`

Figure 8: Coefficients from Poisson regression models with `color` plus an additional predictor. In all four cases, the additional predictor is not significant at the 0.05 level.

Using a similar line of reasoning as before, likelihood ratio tests and AIC can be used to compare the reduced model `spine = color` with every other model of the form `spine = color + ?`. This is presented in Table 3, all proposed models with an additional predictor fails to pass the likelihood ratio test, and yields a worse AIC score.

Model		Log-likelihood		AIC		LR test: P-value
Reduced	Proposed	Reduced	Proposed	Reduced	Proposed	
spine = color	spine = color + weight	-260.41	-260.26	528.82	530.52	0.58
spine = color	spine = color + width	-260.41	-260.39	528.82	530.78	0.83
spine = color	spine = color + sat	-260.41	-260.41	528.82	530.82	0.93
spine = color	spine = color + y	-260.41	-260.34	528.82	530.69	0.71

Table 3: Likelihood-ratio tests and AIC values of each all models with of the form `spine = color + ?`. In all four cases, the proposed model with additional predictor fails to pass the likelihood-ratio test. Also, all proposed models have a worse AIC score compared to `spine = color`

3.1.4 Adding interactions to the model

At this point, it would seem difficult to improve upon `spine = color` just by adding main effects to the model. In this study, some attempts of using models with interactions have been made. The most notable one is `spine = weight + weight:color`. The coefficients are this model are presented in Figure 9.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.05	0.216	4.86	1.15e-06
weight	-0.297	0.126	-2.35	0.0187
weight:color2	0.232	0.0994	2.33	0.0198
weight:color3	0.279	0.104	2.68	0.00729
weight:color4	0.294	0.113	2.61	0.00895

Figure 9: Coefficients from `spine = weight + weight:color`. All predictors in the model are significant at the 0.05 level.

This model has an AIC value of 530.51. The model was tested against the null model `spine = 1` and the reduced model `spine = weight` using likelihood ratio tests. In both cases, `spine = weight + weight:color` is the better model at 0.05 significance level.

Since the predictor of `spine = color` is not present in `spine = weight + weight:color`, these two models cannot be compared using a likelihood-ratio test. In terms of AIC, `spine = color` is better (528.82). But one would wonder if `spine = weight + weight:color` would perform better in terms of predictive power, since AIC also penalizes based on the number of parameters in the model. This issue will be addressed later in Section 3.3.2.

3.2 Automated model selection methods

In this section, some automated model selection techniques that was used in the study will be discussed. These methods rely heavily on R packages, and the chosen model will only be best in terms of AIC or BIC. Nonetheless, they are still very useful for identifying potentially important interactions that were not considered in manual fitting.

3.2.1 Stepwise selection

The stepwise selection methods are *greedy* in nature, because at each iteration, the algorithm proceeds in the direction which yields best immediate improvement (in terms of AIC).

It is a computationally efficient algorithm, but does not guarantee convergence towards the optimal model.

In this study, the `stepAIC()` function from `MASS` package was used for forward selection, backward elimination and forward-backward selection. In each case, the set of relevant predictors needs to be defined to limit the scope of models under consideration. The set of relevant predictors specified was one which includes all first, second order and third order terms, ie. all predictors within `(sat + y + weight + width + color)^3`. Based on this scope, models such as `spine = weight + weight:color` was also considered.

Remarkably, all three stepwise selection techniques returns the final model `spine = color`.

3.2.2 Exhaustive enumeration

Exhaustive enumeration is essentially a brute force approach to the model selection problem. Under a specified set of relevant predictors, each possible combination of them are use to fit a model and compute the corresponding AIC. It is not an elegant approach, because the computational cost grows exponentially with the number of predictors. Nonetheless, the returned final model is guaranteed to be optimal from a specified scope.

For this purpose, the `glmulti()` function from `glmulti` package [1] was used. The set of relevant predictors was specified to be *main effects* plus all possible pairwise interactions. The search space could not be expanded further due to limitations of the current implementation.

The `glmulti()` function was executed twice, one with AIC as the comparison criteria, the other with BIC as the comparison criteria. In both cases, `spine = color` emerged as the optimal model.

3.3 Additional metrics for comparison

The analyses in section 3.2 implies that `spine = color` is indeed the best model — in terms of AIC. But one could also argue that models such as `spine = weight + weight:color` (see figure 9) may have better predictive power due to higher complexity. In this section, alternatives to AIC and BIC that were used to compare `spine = color` with other models are presented.

3.3.1 Pseudo R-squared

The pseudo R-squared of a Poisson regression is defined as

$$R^2 = \frac{\log(\mathcal{L}(\hat{\beta}_0)) - \log(\mathcal{L}(\hat{\beta}))}{\log(\mathcal{L}(\hat{\beta}_0))}$$

where $\mathcal{L}(\hat{\beta}_0)$ is the likelihood of the null model. The pseudo R-squared of a model can be any value from 0 to 1, with 1 being a perfect fit to the dataset. Table 4 shows the pseudo R-squared values of models discussed so far.

The two models with highest pseudo R-squared are `spine = weight + weight:color` and `spine = sat + y + weight + width + color`, this is not surprising because the pseudo R-squared is purely based on likelihood values and does not penalize the complexity of the model. One can simply include arbitrarily large number of predictors into a model to obtain a perfect pseudo R-squared of 1, but it would not be meaningful because the model has clearly overfitted on the data.

Model	Pseudo-R-squared
spine = sat + y + weight + width + color	0.206
spine = color	0.188
spine = weight	0.024
spine = width	0.013
spine = sat	0.007
spine = y	0.001
spine = color + weight	0.193
spine = color + width	0.189
spine = color + sat	0.188
spine = color + y	0.190
spine = weight + weight:color	0.194

Table 4: The two models with highest pseudo R-squared are `spine = weight + weight:color` and `spine = sat + y + weight + width + color`, this is not surprising because the pseudo R-squared is purely based on likelihood values and does not penalize the complexity of the model (or the number of parameters of the model). It is also important to note that `spine = color` has the highest pseudo R-squared out of all single predictor models.

3.3.2 k-fold cross validation error

The k-fold cross validation error is an estimator of the *true expected error* of a model, ie. how well a model will predict on unseen data. The algorithm to calculate this error has k iterations, with the dataset being split into k folds of equal size at the beginning. On the i^{th} iteration, a model is fitted using all data except the i^{th} fold. The fitted model is used to predict on the i^{th} fold to calculate an *error score*. At the end of the k iterations, the average of all error scores is reported as the k-fold cross validation error.

To conduct a k-fold cross validation, two decisions are required. The first is the choice of k, and the second being the definition of *error function* used to measure the error between predicted value and actual value.

Since the Horseshoe crab dataset used in this study is not large (173 records), a large value of k is selected ($k = 10$) to reduce the bias of the estimate. Since the our response variable Y is a count, an appropriate error function would be the widely used mean squared error, given by

$$\text{MSE}_i = \frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y_j})^2$$

where $\hat{y_j}$ and y_i are the predicted and actual values, n is the size of the i^{th} fold. Table 5 shows the cross validation errors of all models experimented in section 3.1.

There is sufficient evidence to now claim that `spine = color` is indeed the best Poisson regression model for the Horseshoe crab dataset.

3.4 Verifying model assumptions

In this section, tools that were used to verify model assumptions of a Poisson regression model are presented. These results will help to reveal whether `spine = color` suffers from overdispersion or underdispersion.

Model	Raw CV error	Adjusted CV error
spine = sat + y + weight + width + color	0.600	0.588
spine = color	0.572	0.568
spine = weight	0.685	0.679
spine = width	0.696	0.685
spine = y	0.693	0.690
spine = sat	0.696	0.686
spine = color + weight	0.585	0.577
spine = color + width	0.589	0.577
spine = color + sat	0.584	0.572
spine = color + y	0.573	0.571
spine = weight + weight : color	0.585	0.581

Table 5: Cross validation errors of single predictor models, models of the form `spine = color + ?`, model with all main effects, and `spine = weight + weight:color`. `spine = color` has the lowest error score, which means that it is the best model for predicting on unseen data. Judging from the errors of `spine = weight + weight:color`, we can deduce that it most probably suffers from overfitting because its higher complexity does not translate to better prediction on out-of-sample data.

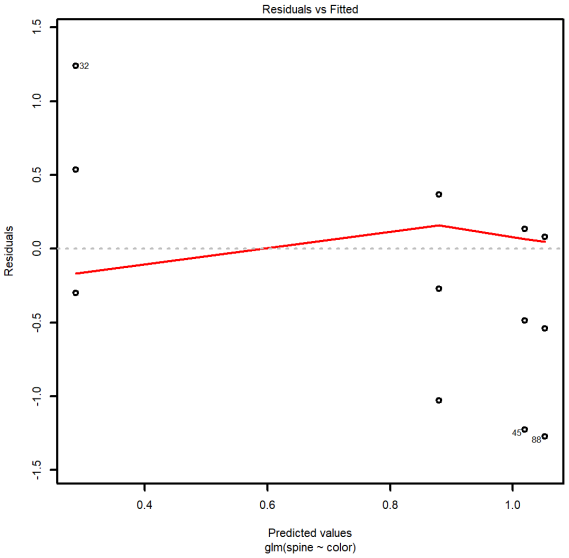
3.4.1 Deviance goodness of fit test

For a Poisson regression model, the deviance statistic

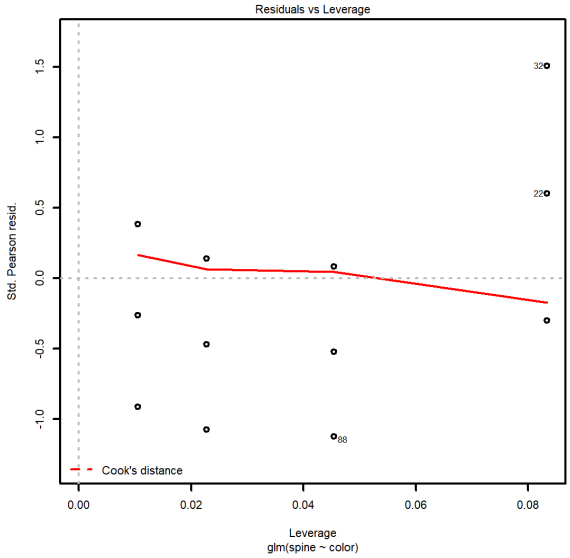
$$D = \sum_{j=1}^{n} [y_i \log(\frac{y_i}{\exp(\beta^T X_i)}) - (y_i - \exp(\beta^T X_i))]$$

is a measure of how closely the model's predictions are to the observed outcomes. If the Poisson model has p parameters, D would follow a chi-square distribution with $n-p$ degrees of freedom. Hence, D can be used to construct a goodness-of-fit with the null hypothesis that the fitted model is correct. In this study, the p-value returned from this hypothesis test is 1, meaning there is insufficient evidence to reject the null, which states that `spine = color` is correctly specified.

3.4.2 Diagnostic plots



(a) Residuals vs fitted plot of `spine = color`. The x-axis is the value of the linear predictor $\beta^T X_i$ for the i^{th} data. The three curvilinear trace of points in this plot shows that response variable Y is discrete and takes on three possible values. The variance in residuals appears to remain relatively constant even as $\beta^T X_i$ increases. This is a sign of underdispersion.



(b) Standard Pearson residuals of `spine = color`. The Standard Pearson residuals corrects for the unequal variance in raw residuals by dividing the standard deviation. Based on model assumptions, it should be normally distributed with 0 mean and constant variance. However, from this plot it appears that there is a quadratic trend.

Figure 10: Dignostic plots of `spine = color`

3.4.3 Dispersion test

Cameron and Travedi [2] developed a test for overdispersion or underdispersion for Poisson regression models. The general framework of the hypothesis test can be described as follows:

$$Var(Y|X) = \mu_{Y|X} + \alpha f(\mu_{Y|X})$$

$$H_0 : \alpha = 0, \quad H_1 : \alpha \neq 0$$

where $\alpha \in \mathbb{R}$, $\mu_{Y|X} = E(Y|X)$, $f(\mu_{Y|X})$ is a specified monotone function (usually quadratic or linear). H_0 is exactly what is assumed by a Poisson model, the variance of $Y|X$ is equal to its mean. This is a two-sided test, $\alpha < 0$ means that underdispersion is occuring, while $\alpha > 0$ means that overdispersion is at work. The test statistic used is a t-statistic which is asymptotically standard normal under the null hypothesis. Its explicit expression is omitted here, and can be found in [reference].

To carry out this test on `spine = color`, the `dispersiontest()` function from `AER` package is used. This implementation supports both one-sided and two-sided versions of the dispersion test described above. The resulting p-values in Table 6 provide strong evidence that `spine = color` is suffering from underdispersion.

H_0	H_1	$f(\mu) =$	Estimated value of α	P-value
$\alpha = 0$	$\alpha \neq 0$	μ	-0.772	2.20e-16
$\alpha = 0$	$\alpha \neq 0$	μ^2	-0.308	2.20e-16
$\alpha = 0$	$\alpha > 0$	μ	-0.772	1
$\alpha = 0$	$\alpha > 0$	μ^2	-0.308	1
$\alpha = 0$	$\alpha < 0$	μ	-0.772	2.20e-16
$\alpha = 0$	$\alpha < 0$	μ^2	-0.308	2.20e-16

Table 6: Resulting p-values from the dispersion test on `spine = color`. The p-values for two-sided dispersion test, and for one-sided underdispersion test are both significant at the 0.05 level, underdispersion is clearly present in the model.

4 Quasi-Poisson regression

In section 3, `spine = color` is being identified as the best Poisson regression model for the Horseshoe crab dataset. However, it was also shown in section 3.4 that the model suffers from underdispersion. This implies that a more flexible model is required to accommodate for this behavior. The quasi-poisson regression model provides a simple solution to this problem. In a quasi-poisson model, an additional parameter θ is added into the model, such that

$$V(Y|X) = \theta E(Y|X) = \theta \exp(\beta^T X)$$

Hence, the Poisson regression is simply a special case of quasi-poisson regression in which $\theta = 1$. θ is known as the ***dispersion parameter***. If $\theta > 1$, then the conditional variance of Y increases more rapidly relative to its conditional mean, in this case the model is fitted to overdispersed data. Conversely, if $\theta < 1$, this means that the model is being fitted to an underdispersed data. If a quasi-poisson model has k parameters, θ is estimated as

$$\hat{\theta} = \frac{1}{n-k} \sum_{j=1}^{n} \frac{(y_i - \exp(\hat{\beta}^T X_j))^2}{\exp(\hat{\beta}^T X_j)}$$

The estimated coefficients $\hat{\beta}$ of a model does not change when moving from a Poisson model to a quasi-Poisson model (everything else held equal). The only change is that the *standard errors* of the estimated coefficients will be scaled based on the estimated value of the dispersion parameter θ. This affects the statistical significance (p-values) of the estimated coefficients.

Since the estimated coefficients remain unchanged, the estimated mean $\exp(\hat{\beta}^T X_i)$, which is used as the prediction of y_i, does not change. This implies that the k-fold Cross-Validation errors (Table 4) should remain unchanged as well when moving from a Poisson model to quasi-Poisson model. We can therefore expect `spine = color` to be the best quasi-Poisson model in terms of k-fold cross validation error.

However, the quasi-Poisson model is not a full maximum likelihood model, but a quasi-maximum likelihood model, so AIC cannot be calculated (`glm` package does not support it). This means that automated stepwise selection methods such as `stepAIC` cannot be used.

4.1 `spine = color` with quasi-Poisson family

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.288	0.121	2.38	0.0184
color2	0.592	0.125	4.74	4.58e-06
color3	0.732	0.129	5.7	5.35e-08
color4	0.764	0.135	5.65	6.76e-08

Figure 11: Coefficients of quasi-Poisson regression model `spine = color` fitted to dataset. All predictors are significant at the 0.05 level, including the intercept. This is an improvement from the Poisson version of `spine = color` (see figure 7). Although the estimated coefficients remain unchanged, the standard errors associated with each coefficient is much lower. This is makes sense because it has been shown in section 3.4.3 that the Horseshoe crab dataset is underdispersed — meaning the Poisson model for `spine = color` *overestimates* the variance of the estimates of coefficients.

Figure 11 shows the summary of coefficients of the quasi-Poisson model `spine = color`. As expected, the estimates of the coefficients, $\hat{\beta}$ remains the same as Poisson variant of `spine = color`. However, the p-values $\hat{\beta}$ of the quasi-Poisson model is much lower, this is a notable improvement.

The dispersion parameter, $\hat{\theta}$ of this model was estimated to be 0.234 (< 1), which is not surprising because the dataset is underdispersed. Recalling the results of one-sided underdispersion test discussed in illustrated in Table 6, the estimated value of α, where $Var(Y|X) = \mu_{Y|X} + \alpha(\mu_{Y|X})$ was found to be -0.772. From $Var(Y|X) = \mu_{Y|X} + -0.722(\mu_{Y|X}) = 0.278\mu_{Y|X}$ which is not very far from the dispersion parameter $\hat{\theta}$ estimated by the quasi-Poisson model. This slight discrepancy is most likely due to difference in methods of estimation, the dispersion test developed by [2] employs regression-based techniques, while the dispersion parameter $\hat{\theta}$ is simply estimated based on the Pearson statistic

$$\sum_{j=1}^{n} \frac{(y_i - \exp(\hat{\beta}^T X_j))^2}{\exp(\hat{\beta}^T X_j)}$$

of a Poisson model.

Based on the results above, it is clear that the quasi-Poisson `spine = color` is a better model compared to the Poisson `spine = color`. In the next section, negative binomial regression models will be tested on the dataset.

5 Negative Binomial regression

The negative binomial distribution is discrete, positive-valued, and belongs to the exponential family. Thus, it fulfills equation (1) and therefore it is a feasible option for modeling the Horseshoe crab dataset.

The negative binomial regression model is best suited for count dataset with overdispersion. This is because, in a negative binomial distribution, there is an additional dispersion parameter $\gamma > 0$ such that the variance can be written as

$$Var(Y) = \mu_Y + \frac{1}{\gamma}{\mu_Y}^2$$

For all $\gamma > 0$, $Var(Y) > \mu_Y$, which is exactly how an overdispersed response variable Y would behave. In fact, the larger γ gets, the closer the distribution of Y approximates the Poisson distribution. Formally, as $\gamma \to \infty$, we have $\text{NB}(\gamma, \frac{\lambda}{\lambda+\gamma}) \to \text{Poisson}(\lambda)$.

Based on this reasoning, it is expected that the negative binomial model will not be a good representation of the Horseshoe crab dataset, which is underdispersed.

5.1 `spine = color` with negative binomial family

The `spine = color` model was fitted using the `nb.glm` function in the `MASS` package. Figure 12 summarizes the estimated coefficients of the model.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.288	0.25	1.15	0.25
color2	0.592	0.259	2.29	0.022
color3	0.732	0.266	2.75	0.00589
color4	0.764	0.28	2.73	0.00632

Figure 12: Coefficients of negative binomial `spine = color` fitted to dataset. Coefficients of `color` are significant at the 0.05 level. The model has an AIC of 530.82.

The dispersion parameter was estimated to be 183942.7, a very large number. Since every negative binomial regression model is overdispersed, given an underdispersed dataset, the best possible fit would be achieved by setting γ to be as large as possible.

Based on this observation, it is safe to conclude that negative binomial models are not appropriate for representing the Horseshoe crab dataset.

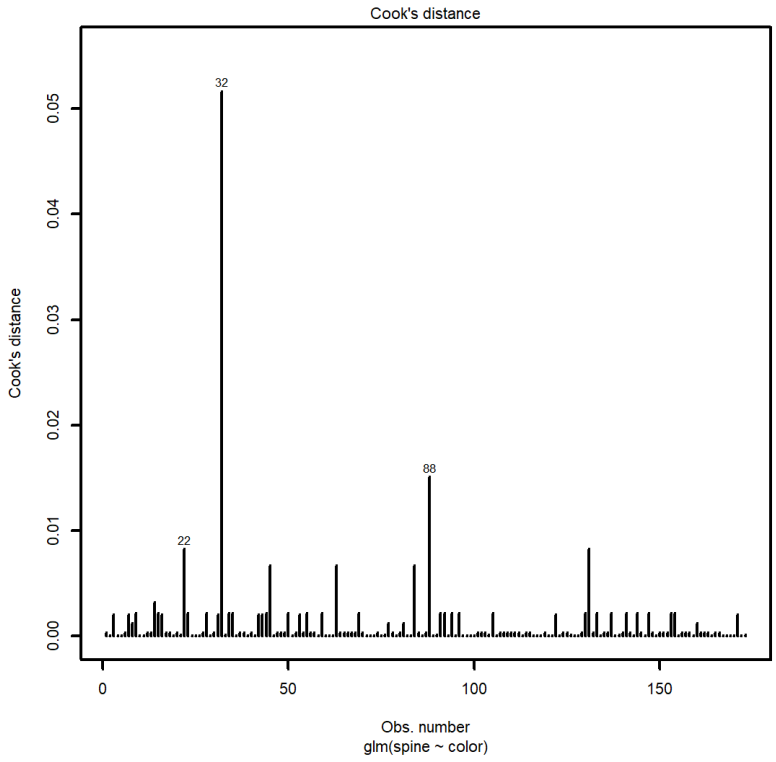
6 Selecting the best model

Based on the findings from section 3, 4, and 5, it is evident that the quasi-Poisson `spine = color` is the superior model. Negative binomial regression models are inappropriate because they are not suited to model underdispersed count data, which is the case the for Horseshoe crab dataset. The quasi-Poisson `spine = color` is slightly better than the poisson `spine = color` because it accounts for the underdispersion of the Horseshoe crab dataset, which leads to more significant p-values for the estimates of ceofficients.

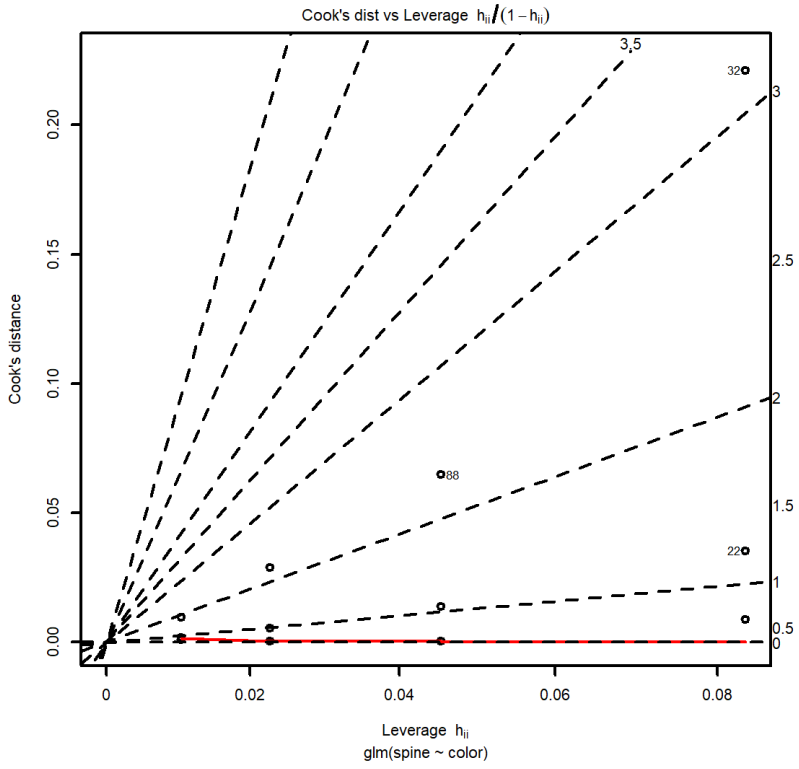
7 Final adjustments: Finding outliers

In the previous section, it is finally established that the `spine = color` quasi-Poisson model is the optimal choice among all other models, including negative binomial models and Poisson models. In this section, the goal is to examine whether there are outliers within the dataset that significantly impair the performance of the `spine = color` quasi-Poisson model.

7.1 Cook's distance plots



(a)



(b)

Figure 13: Cook's distance plot of the quasi-Poisson `color = spine`

Based on the plots in figure 13, one can easily spot the 32nd data point as a highly influential record in terms of model fitting. One of the commonly used Cook's distance threshold to decide whether a record is an outlier is $\frac{4}{N}$, where N is the number of records in the dataset. Following this convention, the threshold is $\frac{4}{173} = 0.0231$. The Cook's distance of the 32nd point is at least two times this threshold, thus there is sufficient evidence to consider it as an outlier.

Refitting the the quasi-Poisson `color = spine` without the 32nd record, we would obtain the following summary of coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.167	0.131	1.28	0.203
color2	0.713	0.135	5.3	3.61e-07
color3	0.853	0.138	6.2	4.32e-09
color4	0.885	0.144	6.16	5.24e-09

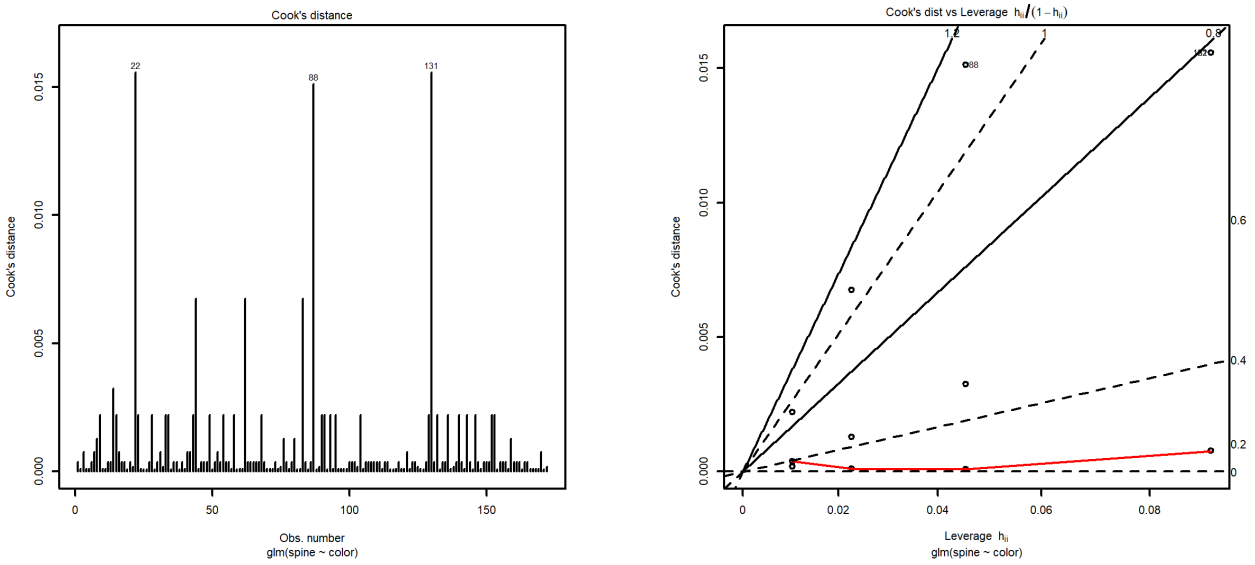
Figure 14: Coefficients of quasi-poisson `spine = color` fitted to the dataset with 32nd record removed. It can be noted that the estimated coefficients have changed significantly compared to the previous model trained with the original dataset (see figure 11).

Another way to visualize why the 32nd record is such an influential point is by simply looking at the values in this particular record. The 32nd record in the original Horseshoe crab dataset is:

- `sat = 0, y = 0, weight = 2.6, width = 25.8, color = 1`
- `spine = 3`

Referring back to the histogram plots of `spine` with respect to `color` presented in section 2.3 (see figure 3), we can see that there is only one record in the entire dataset where `spine` takes the value of `3` when `color` is `1`, which is precisely the 32nd record. Although similar extreme cases can also be seen when `color = 4`, the 32nd record can be expected to be much more influential on the estimated coefficient corresponding to the indicator variable of `color = 1` (ie. the intercept, see figure 14) there are only a small number of records (12) with `color = 1`.

By removing the 32nd data point, the k-fold Cross Validation error of the model improves significantly, decreasing from 0.572 to 0.546. The dispersion parameter $\hat{\theta}$ decreases from 0.234 to 0.223, indicating the dataset without 32nd record is slightly more underdispersed. The Cook's distance plot of this refitted model (figure 15) shows that there are still several potentially influential records in the dataset, but there is no longer one specific record that stands out significantly.



(a) **(b)**

Figure 15: Cook's distance plots of the quasi-Poisson `color = spine` fitted on Horshoe crab dataset without 32nd record. There are still several potentially influential records in the dataset, but there is no longer one specific record that stands out significantly, thus this is an acceptable result.

8 Conclusion

In this study, the goal is to identify a generalized linear model that can predict `spine` in the Horseshoe crab dataset using `color, y, sat, width, weight`. Before modeling `spine`, extensive exploratory data analysis was carrierd out on dataset (section 2). Some of the visualization produced during this step proved to be extremely helpful in the latter stages of modeling. In particular, the histogram plots of `spine` with respect to `color` (see figure 3) revealed the importance of `color` in predicting `spine`.

A large amount of effort in this study was devoted to identifying the best possible Poisson model for predicting `spine` (section 3) before proceeding to more flexible models such as Negative Binomial models and Quasi-Poisson models. Many tools were used to compare different Poisson models, such as the AIC, BIC, likelihood ratio test, pseudo R-squared, and k-fold cross validation error. Automated model selection techniques was also used to efficiently explore a large set of candidate models. It was then established that `spine = color` is the best Poisson model in terms of AIC, BIC and k-fold cross validation error.

Then, diagnostic plots, goodness-of-fit deviance test and dispersion test was used to verify the Poisson model assumptions. It was then discovered that the Horseshoe crab dataset is underdispersed, meaning the `spine = color` Poisson model overestimates the variance of the response variable `spine`.

Using the valuable information gained in Poisson modeling, it was relatively easy to transition to the more flexible quasi-Poisson and Negative Binomial models. The quasi-Poisson `spine = color` was found to be a better fit to the dataset compared to the poisson `spine = color`, judging from the smaller p-values of the estimated coefficients (figure 11). Negative binomial models were quickly regarded as inappropriate for this study after an experimentation because they are only suitable for modeling overdispersed count data, ie. $Var(Y|X) > E(Y|X)$.

Based on these results, the quasi-Poisson `spine = color` was regarded as the optimal model. Finally, in section 7, Cook's distance plots were used to identify possible outliers

in the Horseshoe crab dataset. The 32nd record was removed due to its high influence on the estimated coefficients of `spine = color`.

8.1 Future work

Although the quasi-Poisson `spine = color` was regarded as the optimal model in this study, there are certainly many models of other distributions that can be superior. For instance, the Conway Maxwell Poisson Regression model [3] is a highly flexible regression model for count data that is able to account for underdispersion as well. In this study, the `COMPoissonReg` package in R was briefly used to explore the capabilities of the Conway Maxwell Poisson Regression. However, due to time constraint, no conclusive findings were obtained. In the future, it may be fruitful to devote time in exploring this class of models.

References

[1] Vincent Calcagno and Claire De Mazancourt. glmulti: Anrpackage for easy automated model selection with (generalized) linear models. *Journal of Statistical Software*, 34(12), 2010.

[2] A.colin Cameron and Pravin K. Trivedi. Regression-based tests for overdispersion in the poisson model. *Journal of Econometrics*, 46(3):347–364, 1990.

[3] Kimberly F. Sellers, Sharad Borle, and Galit Shmueli. *Applied Stochastic Models in Business and Industry*, 28(2):104–116, Aug 2011.