Final Report for Data Modeling Project

Annalise Chang, George Guo, Nicholas Way

STAT 400: Statistical Modeling II

Table of Contents

Chapter 1: Logistic Modeling with Poisonous Mushrooms

To begin our investigation into which variables are best at predicting whether a mushroom is poisonous, we analyzed a data set of 8,124 mushrooms from the Audubon Society Field Guide. These data were collected in 1987 and are now distributed through the UC Irvine Machine Learning Repository. A sample of the cleaned data is available in Appendix A. The data included 22 categorical variables describing various features of the mushrooms. These details included surface textures, colors, odors, and location features, among other things. The identified response variable was whether a mushroom was known to be poisonous or not. As such, we aimed to identify which of a selection of variable clusters was superior in predicting the poison status of a mushroom with logistic models.

Exploratory Data Analysis

To begin, we needed to convert the data set into a more usable format. Specifically, we renamed the variables and recode their values according to a code book. Originally, the data were presented with nameless columns and values were encoded to single letters. In renaming the data, we became more familiar with its structure and what to expect when we formally began the exploratory data analysis. Some examples of the renaming code are provided in Appendix A.

Once the data set was in order, we analyzed features of the data. The set did not have any unintentional missing data, but one of the variables (which described the root structure of the mushroom) had a missing value indicator deliberately set in the codebook. Other than that variable, there were no variables that contained missing values. Next, we set to investigate the variables and possible relationships, especially those between the predictor variables and poison status grouping. We began with generating frequency tables for each variable and its levels, then produced faceted bar plots to investigate frequency of possible predictor variables between poisonous and non-poisonous mushrooms. Example code for the plots is in Appendix A.

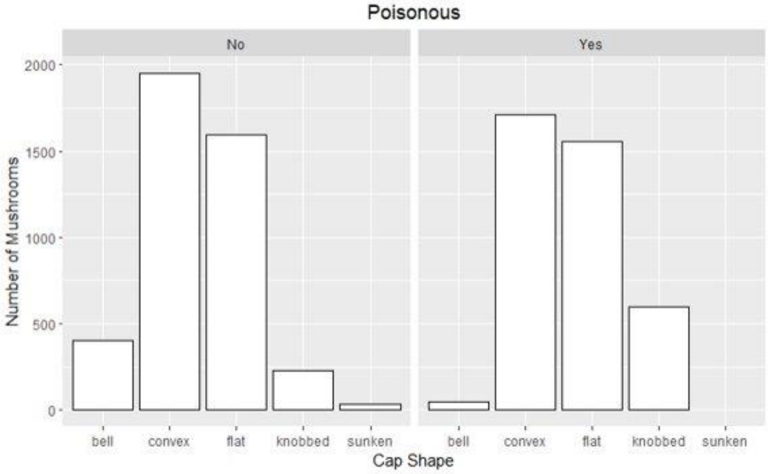


Figure 1: Bar plot of cap shape faceted by poison status.

After investigating individual frequencies, we decided to drop one of the variables due to every case having the same value. This left us with 21 predictor variables plus the poison status indicator. In Figure 1, we have an example of the plots we generated. In it, we investigate the cap shape variable, which describes the shape of the top of the mushroom. There are five possible levels: bell, convex, flat, knobbed, and sunken. As Figure 1 indicates, there is little apparent variation in the cap shape distribution between non-poisonous and poisonous mushrooms.

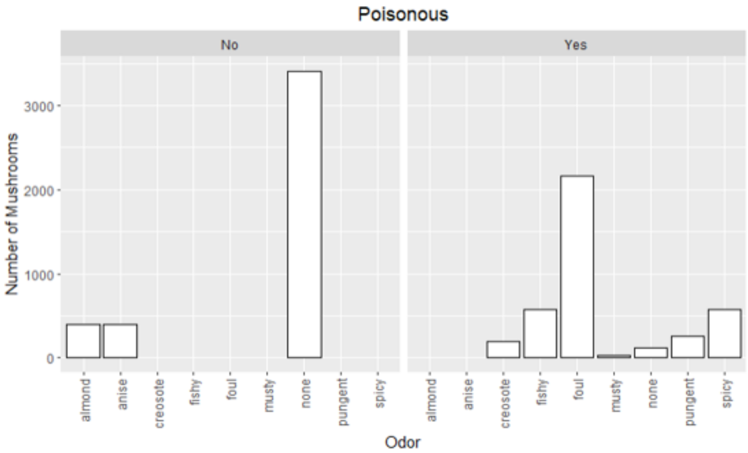


Figure 2: Bar plot of odor faceted by poison status.

When considering the odor variable, which describes the smell that the mushroom gives off, we have a more interesting plot. Figure 2 shows the faceted bar plot associated with the odor variable. The figure indicates that non-poisonous mushrooms overwhelmingly tend to have no odor, while poisonous mushrooms usually have some sort of smell, typically a bad one.

Few plots had as clear a grouping as that for odor. However, many exhibited moderate potential differences, while others were more consistent between poisonous and non-poisonous mushrooms, much like the plot for cap shape. With a stronger understanding of the variables and how they may relate to the response variable, we moved on to fitting and comparing models.

Logistic Model Fitting and Selection

Intuitively, the data appeared to cluster itself into eight categories: cap, odor, gills, stalk, color below cap, rings, spores, and environment. We decided to create logistic models using these clusters instead of using a formal feature selection method to provide us with many models to compare with one another. Additionally, this decision had some intuitive reasoning: modeling by specified clusters allows us to make conclusions that translate smoothly to practice. If one model is superior to the others, we may say that if you run into a mushroom, you may best be able to determine its poison status by focusing on one part of the mushroom, such as its cap or stalk. In reality, the rule of thumb is to assume all wild mushrooms are poisonous unless definitively proven otherwise, so we would not recommend using any generated models to inform a mushroom foraging session.

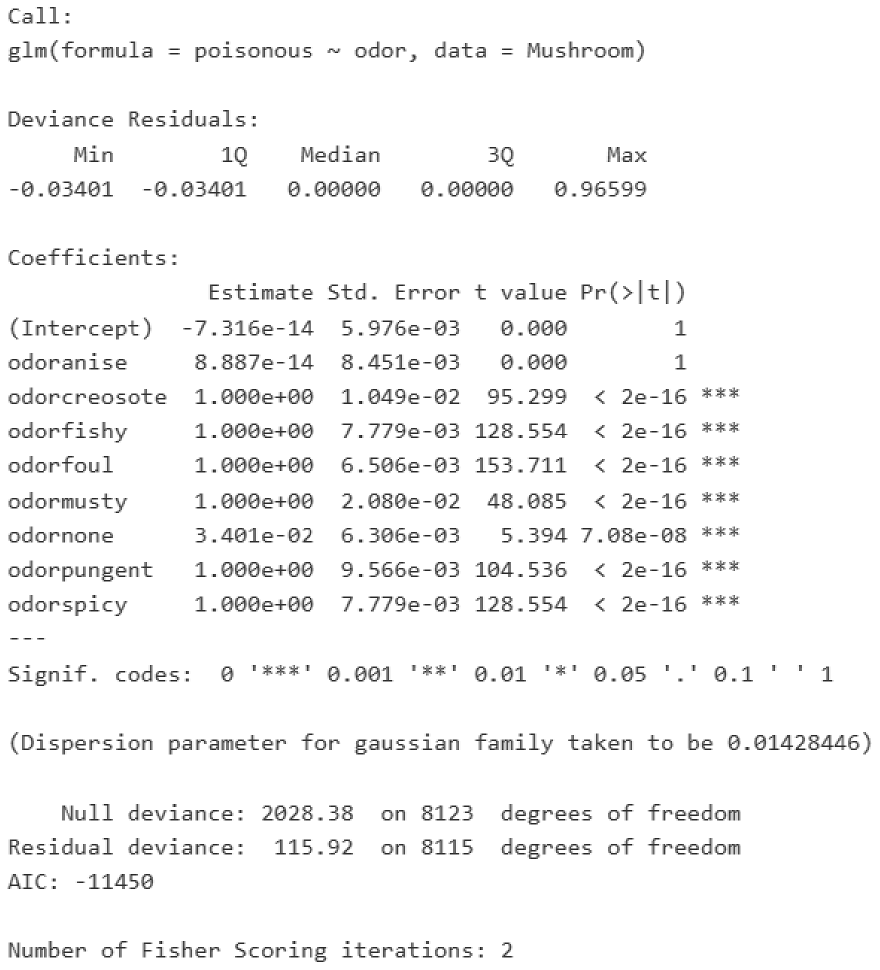
Sample code used to generate models and comparison parameters is included in Appendix A. Table 1 displays certain parameters for each model: Akaike information criterion (AIC), Baysan information criterion (BIC), accuracy, and the area under a receiver operating characteristic (ROC) curve. The first two parameters allow us to compare the model parameter likelihoods, and the last two are related to the predictions that each logistic model makes. In general, we are looking for a model with minimal AIC and BIC for model selection. However, we also want to consider accuracy and area under the curve to understand the prediction strength of our selected model.

Table 1: AIC, BIC, Accuracy, and AUC values for each model.

Model	AIC	BIC	Accuracy	AUC
Cap	7867.9	8000.963	0.806	0.868
Odor	**-11450**	**-11379.56**	**0.9852**	**0.9876**
Gill	4381.2	4493.247	0.8705	0.9436
Stalk	485.87	558.8889	0.9135	0.965
Non-cap Color	8317.2	8436.287	0.7676	0.8384
Ring	7909.6	7958.58	0.7991	0.8295
Spore	5012.1	5082.118	0.868	0.8983
Environment	8279.6	8370.675	0.7543	0.8455

As the bolded row of values on Table 1 indicates, the preferred model is that which analyzes the odor of a mushroom. If you recall Figure 2 in the exploratory data analysis, this outcome makes sense; there were distinct groupings of odor types by poison status. The AIC and BIC values agree across the models, with those for the odor model being substantially lower than the values for the other models. Furthermore, the odor model boasts a strong accuracy and an area under the curve close to 1. These indicate that the predictions that the odor model makes are strong within the data set.

Figure 3 displays the summary for the odor model. While most p-values related to the predictor variable levels were significant, that for an anise odor was not. Additionally, the estimated parameters for all coefficients lie at essentially one or close to zero. That is to say that the model looks at a mushroom’s odor and assigns it as poisonous or not based on which odor it has. Given that we are dealing solely with categorical variables, this classification method makes sense. Using this interpretation, we understand that the model predicts that mushrooms with anise or no odor will not be poisonous, but mushrooms with creosote, fishy, foul, musty, pungent, or spicy odors will be poisonous. This model is also advantageous over others in its parsimony; this model had a low predictor indicator variable count relative to the other fitted models.



```
Call:
glm(formula = poisonous ~ odor, data = Mushroom)

Deviance Residuals: 
     Min        1Q    Median        3Q       Max  
-0.03401  -0.03401   0.00000   0.00000   0.96599  

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -7.316e-14  5.976e-03   0.000        1    
odoranise     8.887e-14  8.451e-03   0.000        1    
odorcreosote  1.000e+00  1.049e-02  95.299  < 2e-16 ***
odorfishy     1.000e+00  7.779e-03 128.554  < 2e-16 ***
odorfoul      1.000e+00  6.506e-03 153.711  < 2e-16 ***
odormusty     1.000e+00  2.080e-02  48.085  < 2e-16 ***
odornone      3.401e-02  6.306e-03   5.394 7.08e-08 ***
odorpungent   1.000e+00  9.566e-03 104.536  < 2e-16 ***
odorspicy     1.000e+00  7.779e-03 128.554  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.01428446)

    Null deviance: 2028.38  on 8123  degrees of freedom
Residual deviance:  115.92  on 8115  degrees of freedom
AIC: -11450

Number of Fisher Scoring iterations: 2
```

Figure 3: Model summary output for odor model.

However, it is important to note that such a model violates the linearity assumption of logistic models, given that all predictors are categorical. If one were to create a logit plot, it would be difficult to establish linearity due to the binary nature of the model and the predictor variables' categorical nature. Note, however, that this issue would persist for any model made from these data, as all potential variables are categorical.

Regardless, informed by the model comparison parameters, we conclude that a mushroom’s odor alone is best at predicting its poison status relative to models that utilize cap, gill, stalk, color, ring, spore, and environment features. Again, this model has risks if used as the sole predictor of poison status, but it may be an accurate starting point to categorize mushrooms.

Chapter 2: Multilevel Modeling with Reaction Time

We analyzed data collected in a driving reaction time study to answer the question, *"Does the interaction between the salience of an image change and a participant's age impact the amount of time it takes them to detect and react to it?"* The data initially consisted of 7803 observations of 14 variables. 157 of the trials did not result in a recorded result, leaving 7646 complete cases. A sample of the data is available in Appendix B. Participant information included their sex, age, and two variables about age cutoffs and relative ages. Trial information included how meaningful an image was to driving, how noticeable the change was in the picture, and those values re-centered at zero. Results of the study were recorded as a participant's reaction time in seconds and the logarithm of that value.

Exploratory Data Analysis

To begin, we investigated the response variable: reaction time. Since the data may not be independent, in that there are multiple observations for one participant, we aggregated the data by participant identifier to produce the histogram in Figure 4. It indicates that most participants had a mean reaction time across trials of about five seconds. Sample code for aggregated plots like Figure 4 is included in Appendix B.

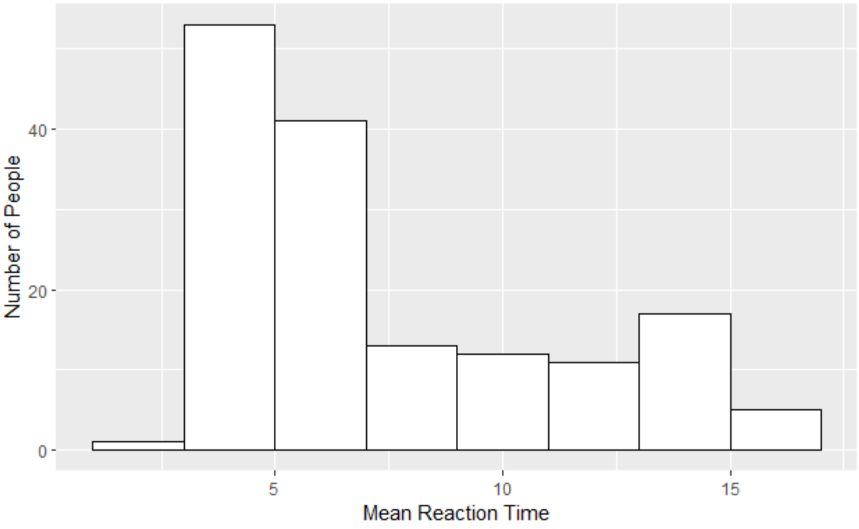


Figure 4: Histogram of mean reaction time, in seconds, aggregated by participant.

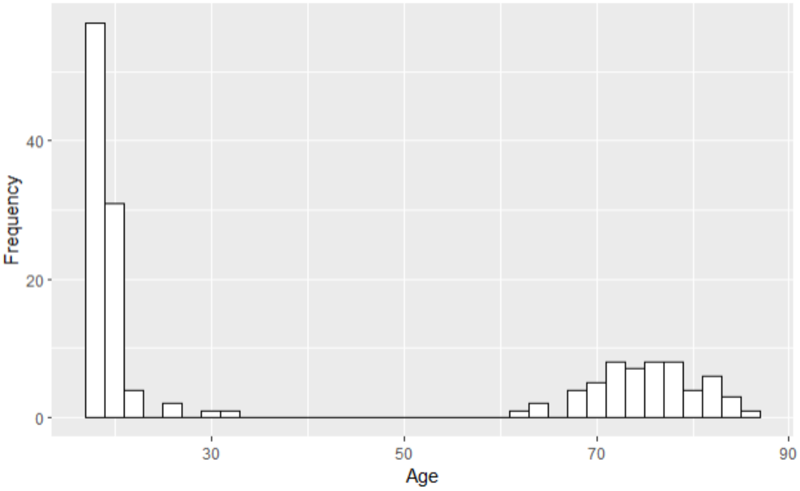


Figure 5: Histogram of participant age, in years.

When investigating participant age, there was an interesting pattern that arose. Figure 5 shows a histogram of participant ages and clearly displays a bimodal distribution. One of the groups clusters around the early twenties, and the other group lies more in the seventies range. The younger age group may be due to college student research participation, but no definitive reasoning is provided. Thus, it may only be reasonable to predict values in these two age ranges with the resulting model and not ages in between.

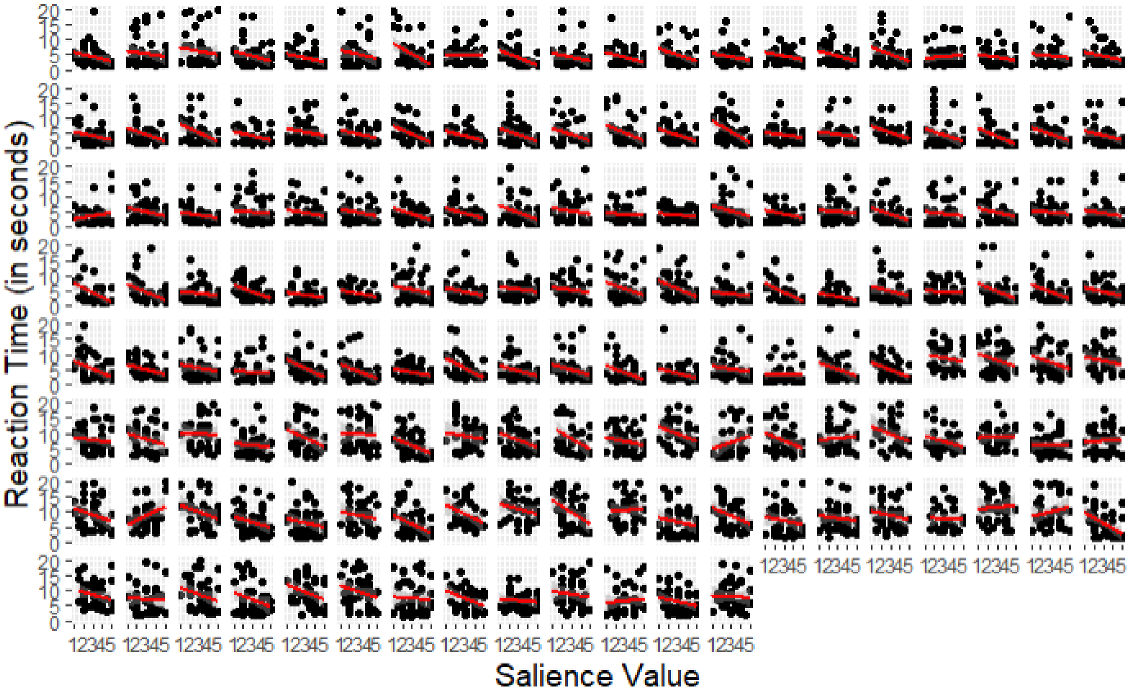


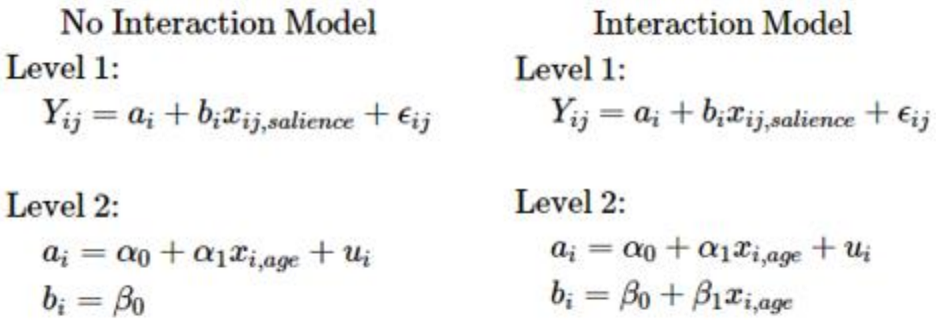
Figure 6: Lattice plot of the relationship between salience and reaction time by participant.

Figure 6 is a lattice plot demonstrating the relationship between the salience value of images and reaction time separated by participant identifier. That is, each participant has their own graph in the figure. The red lines indicate the potential linear directionality, and it is apparent that the slopes of these lines are different between participants. Figure 6 enforces the idea that a multilevel model is appropriate for this data, as it displays an influence of within-person variability on reaction times. One may note that the relationships are predominantly negative, meaning that as image salience increases and an image is more different than a previous, a person's reaction time to the change typically decreases. Sample code for this lattice plot is included in Appendix B.

To further explore the appropriateness of a multilevel model for the data, we constructed a random intercepts model to calculate the intraclass correlation coefficient. The code to produce this model is included in Appendix B. After developing this model and calculating the coefficient, we determined that approximately 18.72% of the total variation in reaction times come from between-participant differences. We continued our analyses acknowledging that the intraclass correlation provided weak evidence in favor of using a multilevel model.

Multilevel Model Fitting and Conclusions

To answer our research question, we had to build two multilevel models: one with an interaction between age and salience and one without. The multilevel model form of each is included in Figure 7. The code used to fit and later compare these models is included in Appendix B. Note that the error term for the b_i equation is omitted in both models to reduce complexity. The resulting composite models differ only by the interaction term; they contain the same variance structure.



No Interaction Model

Level 1:

$$Y_{ij} = a_i + b_i x_{ij,salience} + \epsilon_{ij}$$

Level 2:

$$a_i = \alpha_0 + \alpha_1 x_{i,age} + u_i$$
$$b_i = \beta_0$$

Interaction Model

Level 1:

$$Y_{ij} = a_i + b_i x_{ij,salience} + \epsilon_{ij}$$

Level 2:

$$a_i = \alpha_0 + \alpha_1 x_{i,age} + u_i$$
$$b_i = \beta_0 + \beta_1 x_{i,age}$$

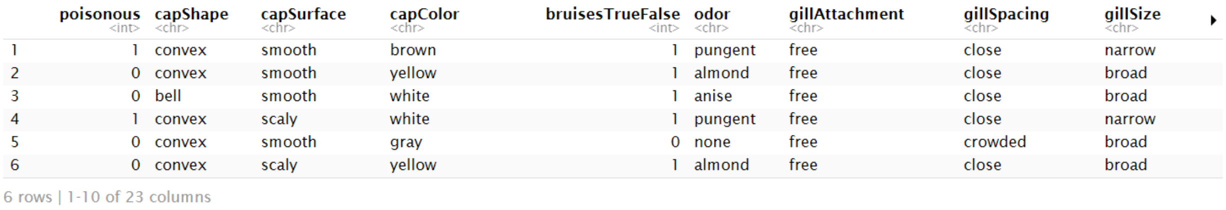
Figure 7: Multilevel model representations for the models of interest.

Since the models are nested, we decided to perform a drop-in-deviance test to compare the two. The hypotheses of this test are as follows: the null hypothesis states that the coefficient term associated with the interaction term equals zero, and the alternative hypothesis states that the term is not zero. In other words, the null hypothesis states that the no-interaction model in Figure 7 is preferred while the alternative hypothesis favors the interaction model. The resulting test statistic was 12.392, which produces a p-value of 0.000431 on a chi-square distribution with one degree of freedom. If we set a desired alpha-level to 0.05, then we may conclude that we reject the null hypothesis that the interaction term has a zero coefficient.

Given our analyses, we conclude that the interaction between the salience of an image and the age of a participant is an important predictor of reaction time in a multilevel model that already contains the separate salience and age variables. In future analyses, it may be productive to investigate other variables included in the data, including the meaning of an image or the distance an older participant is from 65 years old. Furthermore, we may consider more complex models with a greater number of variables.

Appendix A: Data and Code for the Mushroom Models

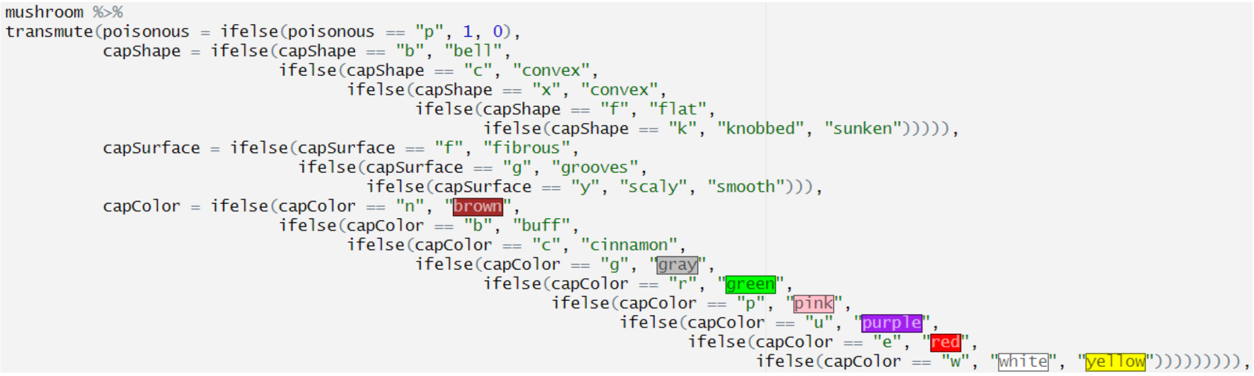
Sample of data set using head function:



	poisonous <int>	capShape <chr>	capSurface <chr>	capColor <chr>	bruisesTrueFalse <int>	odor <chr>	gillAttachment <chr>	gillSpacing <chr>	gillSize <chr>
1	1	convex	smooth	brown	1	pungent	free	close	narrow
2	0	convex	smooth	yellow	1	almond	free	close	broad
3	0	bell	smooth	white	1	anise	free	close	broad
4	1	convex	scaly	white	1	pungent	free	close	narrow
5	0	convex	smooth	gray	0	none	free	crowded	broad
6	0	convex	scaly	yellow	1	almond	free	close	broad

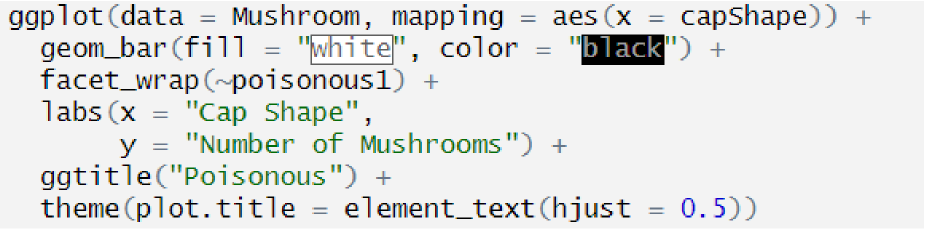
6 rows | 1-10 of 23 columns

Example code lines for renaming data:



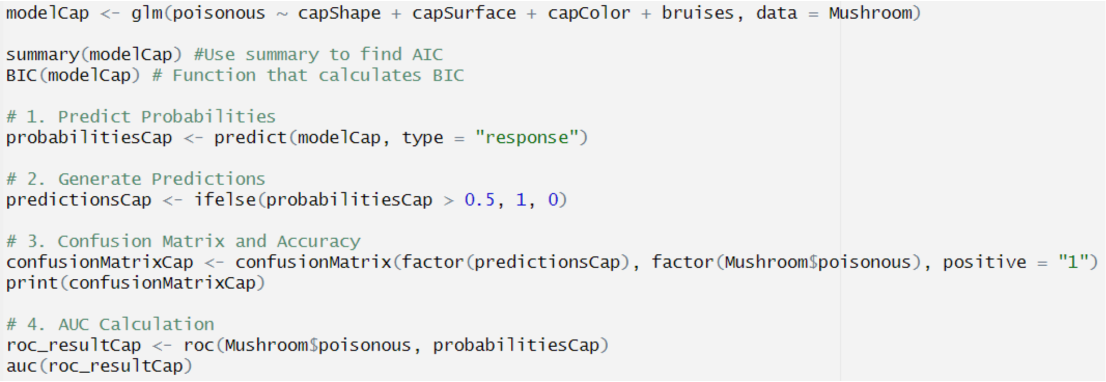
```
mushroom %>%
transmute(poisonous = ifelse(poisonous == "p", 1, 0),
          capShape = ifelse(capShape == "b", "bell",
                            ifelse(capShape == "c", "convex",
                                   ifelse(capShape == "x", "convex",
                                          ifelse(capShape == "f", "flat",
                                                 ifelse(capShape == "k", "knobbed", "sunken"))))),
          capSurface = ifelse(capSurface == "f", "fibrous",
                              ifelse(capSurface == "g", "grooves",
                                     ifelse(capSurface == "y", "scaly", "smooth"))),
          capColor = ifelse(capColor == "n", "brown",
                            ifelse(capColor == "b", "buff",
                                   ifelse(capColor == "c", "cinnamon",
                                          ifelse(capColor == "g", "gray",
                                                 ifelse(capColor == "r", "green",
                                                        ifelse(capColor == "p", "pink",
                                                               ifelse(capColor == "u", "purple",
                                                                      ifelse(capColor == "e", "red",
                                                                             ifelse(capColor == "w", "white", "yellow"))))))))),
```

Example code lines for faceted bar plots:



```
ggplot(data = Mushroom, mapping = aes(x = capShape)) +
  geom_bar(fill = "white", color = "black") +
  facet_wrap(~poisonous1) +
  labs(x = "Cap Shape",
       y = "Number of Mushrooms") +
  ggtitle("Poisonous") +
  theme(plot.title = element_text(hjust = 0.5))
```

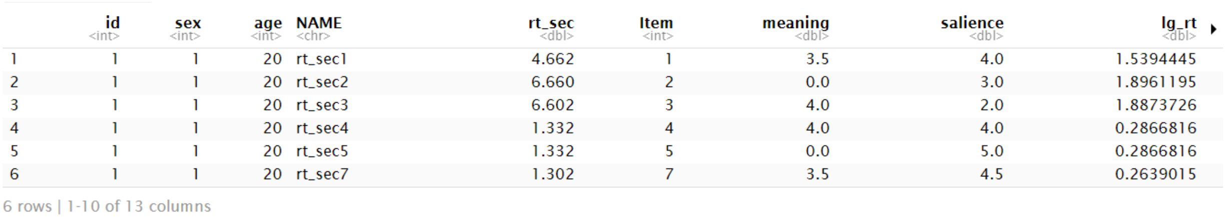
Example code for model generation:



```
modelCap <- glm(poisonous ~ capShape + capSurface + capColor + bruises, data = Mushroom)

summary(modelCap) #Use summary to find AIC
BIC(modelCap) # Function that calculates BIC

# 1. Predict Probabilities
probabilitiesCap <- predict(modelCap, type = "response")

# 2. Generate Predictions
predictionsCap <- ifelse(probabilitiesCap > 0.5, 1, 0)

# 3. Confusion Matrix and Accuracy
confusionMatrixCap <- confusionMatrix(factor(predictionsCap), factor(Mushroom$poisonous), positive = "1")
print(confusionMatrixCap)

# 4. AUC Calculation
roc_resultCap <- roc(Mushroom$poisonous, probabilitiesCap)
auc(roc_resultCap)
```

Appendix B: Data and Code for the Reaction Time Models

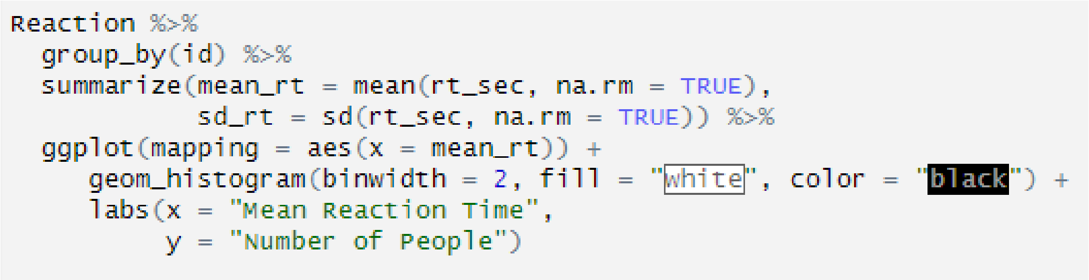
Sample of data set using head function:



	id	sex	age	NAME	rt_sec	Item	meaning	salience	lg_rt
	<int>	<int>	<int>	<chr>	<dbl>	<int>	<dbl>	<dbl>	<dbl>
1	1	1	20	rt_sec1	4.662	1	3.5	4.0	1.5394445
2	1	1	20	rt_sec2	6.660	2	0.0	3.0	1.8961195
3	1	1	20	rt_sec3	6.602	3	4.0	2.0	1.8873726
4	1	1	20	rt_sec4	1.332	4	4.0	4.0	0.2866816
5	1	1	20	rt_sec5	1.332	5	0.0	5.0	0.2866816
6	1	1	20	rt_sec7	1.302	7	3.5	4.5	0.2639015

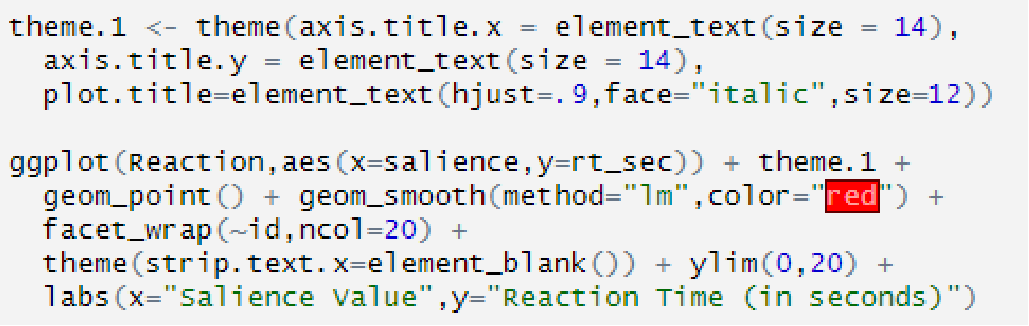
6 rows | 1-10 of 13 columns

Sample code for aggregated plots:



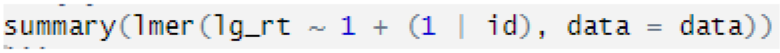
```
Reaction %>%
  group_by(id) %>%
  summarize(mean_rt = mean(rt_sec, na.rm = TRUE),
            sd_rt = sd(rt_sec, na.rm = TRUE)) %>%
  ggplot(mapping = aes(x = mean_rt)) +
     geom_histogram(binwidth = 2, fill = "white", color = "black") +
     labs(x = "Mean Reaction Time",
          y = "Number of People")
```

Sample code for lattice plot:



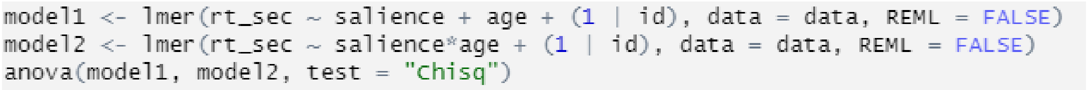
```
theme.1 <- theme(axis.title.x = element_text(size = 14),
  axis.title.y = element_text(size = 14),
  plot.title=element_text(hjust=.9,face="italic",size=12))

ggplot(Reaction,aes(x=salience,y=rt_sec)) + theme.1 +
  geom_point() + geom_smooth(method="lm",color="red") +
  facet_wrap(~id,ncol=20) +
  theme(strip.text.x=element_blank()) + ylim(0,20) +
  labs(x="Salience Value",y="Reaction Time (in seconds)")
```

Sample code for random intercept model generation:



```
summary(lmer(lg_rt ~ 1 + (1 | id), data = data))
```

Sample code for non-interaction, interaction model generation and drop-in-deviance test:



```
model1 <- lmer(rt_sec ~ salience + age + (1 | id), data = data, REML = FALSE)
model2 <- lmer(rt_sec ~ salience*age + (1 | id), data = data, REML = FALSE)
anova(model1, model2, test = "Chisq")
```