1 Summary

The purpose of this project was to determine a model for classifying the edibility or toxicity of mushrooms from a given set of characteristics. Training and test data was taken from the [2], which was originally sourced from the The Audubon Society Field Guide to North American Mushrooms (1981) [1]. The data was processed, screened, and grouped as necessary. Five parametric and non-parametric models were considered for predicting the toxicity classification. The data was discovered to be well-separated, causing a high degree of accuracy in most non-parametric models, and non-convergence in most parametric models. Based on testing, it was concluded that the random forest tree model is the superior model, as it perfectly predicted the data, was easy to interpret, and used fewer total trees than the bagged tree model.

2 Methods

All data processing, analysis, and modeling was performed with the R statistical coding language, with the following code packages:

- `readr` for data input functions
- `dplyr` for data manipulation functions
- `ggplot2` for plotting functions
- `tidyr` for data wrangling functions
- `rpart` for classification tree models
- `ROCR` for ROC plotting
- `randomForest` for Bagging/Random Forest models
- `gbm` for Boosted Tree models
- `class` for K-Nearest Neighbor (KNN) models
- `MASS` for LDA/QDA models

All R code will be provided in a separate file, "Project1.R."

3 Data Sourcing, Screening, and Analysis

3.1 Data Sources

The mushroom data used for this project was retrieved from the open source UCI Machine Learning Repository [2] as the comma-separated value (CSV) format file "mushrooms.csv." The mushroom data was originally prepared by The Audubon Society Field Guide to North American Mushrooms (1981) [1].

3.2 Data Analysis

Appendix A shows the R `summary()` output of the raw "mushrooms.csv" data. To better fit feature labels in the data output, features ending in "...above.ring" or "...below.ring" were shortened to "...ar" and "...br," respectively. The data contains 8124 observations of 22 qualitative features and one classification of "e" for edible and "p" for poisonous. Per the data source [2], the observation features have the following keys:

1. cap-shape: bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s
2. cap-surface: fibrous=f,grooves=g,scaly=y,smooth=s
3. cap-color: brown=n,buff=b,cinnamon=c,gray=g,green=r, pink=p,purple=u,red=e,white=w,yellow=y
4. bruises?: bruises=t,no=f
5. odor: almond=a,anise=l,creosote=c,fishy=y,foul=f, musty=m,none=n,pungent=p,spicy=s
6. gill-attachment: attached=a,descending=d,free=f,notched=n
7. gill-spacing: close=c,crowded=w,distant=d
8. gill-size: broad=b,narrow=n
9. gill-color: black=k,brown=n,buff=b,chocolate=h,gray=g, green=r,orange=o,pink=p,purple=u,red=e, white=w,yellow=y
10. stalk-shape: enlarging=e,tapering=t
11. stalk-root: bulbous=b,club=c,cup=u,equal=e, rhizomorphs=z,rooted=r,missing=?
12. stalk-surface-above-ring: fibrous=f,scaly=y,silky=k,smooth=s
13. stalk-surface-below-ring: fibrous=f,scaly=y,silky=k,smooth=s
14. stalk-color-above-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o, pink=p,red=e,white=w,yellow=y
15. stalk-color-below-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o, pink=p,red=e,white=w,yellow=y
16. veil-type: partial=p,universal=u
17. veil-color: brown=n,orange=o,white=w,yellow=y
18. ring-number: none=n,one=o,two=t
19. ring-type: cobwebby=c,evanescent=e,flaring=f,large=l, none=n,pendant=p,sheathing=s,zone=z
20. spore-print-color: black=k,brown=n,buff=b,chocolate=h,green=r, orange=o,purple=u,white=w,yellow=y
21. population: abundant=a,clustered=c,numerous=n, scattered=s,several=v,solitary=y
22. habitat: grasses=g,leaves=l,meadows=m,paths=p, urban=u,waste=w,woods=d

Feature 11, the stalk root type, has a "?" character representing a missing value. The "?" character was set by R to be read as type NA. Screening NA's from the data set yielded 5644 observations, indicating 8124 - 5644 = 2480 observations with missing data. The observations with missing data will be excluded from preliminary modeling, and will be reincorporated if the `stalk.root` feature is eliminated from the final model. All observations of the `veil.type` feature contain the "p" type, therefore this feature was removed from the analyzed data.

The `odor` feature was grouped into the factor levels "Yes" and "No" to indicate whether or not the mushroom had an odor. Variables with "n" were grouped into the "No" category and all other levels were grouped into "Yes." This will reduce the number qualitative factors the models will have to consider. Grouping is justified here because the classification model output from this project is intended to act as an estimation of mushroom toxicity rather than a precise model. More precise evaluations of mushroom toxicity can be performed from either laboratory analysis or looking the mushroom species up from a known listing.

The `ring.number` feature is a numeric feature in the space [0, 1, 2] represented by the factors ["n", "o", "t"]. The feature was changed to a numerical type.

4 Modeling

The following classification models were considered:
1. Classification Tree(s)
2. Logistic
3. Linear Discriminant Analysis (LDA)
4. Quadratic Discriminant Analysis (QDA)
5. K-Nearest Neighbor (KNN)

4.1 Classification Tree(s)

A classification tree and subsequent ensemble models were first considered because of the qualitative nature of the features. The features can be more easily divided without the use of dummy variables, which can reveal initial patterns in the data and provide insight on which variables hold greater predictive value.

4.1.1 Single Classification Tree

The cleaned data observations from Section 3.2 were randomly separated approximately 75% into training data set and approximately 25% into a test set. The `rpart` function was used to fit the classification tree model on the training data. The complexity parameter `cp` was set to the arbitrarily low value of 0.00001 to maximize the size of the initial tree. The `summary()` command on the tree object calculated the following list of variable importance:

```
Variable importance
spore.print.color  stalk.surface.ar  stalk.surface.br         ring.type
               21                15                15                15
   stalk.color.ar    stalk.color.br         gill.size           habitat
               10                10                 5                 3
      stalk.shape         cap.color              odor         cap.shape
                2                 2                 1                 1
      ring.number        population
                1                 1
```

Because the `stalk.root` feature was not identified as important, the records omitted in Section 3.2 containing NA's were added back to the analyzed data set, with the `stalk.root` column removed. The classification tree described above was refit to the new data. Figure 1 shows the resulting tree and 10-fold cross validation as a function of complexity parameter and tree size.

Figure 1 shows that while the 10 node model has the lowest cross validation error, it is within 1 standard deviation of the 7 node model. To minimize model variance, the 7 node model with a complexity parameter of 0.005 will be used.

The 7 node model (Figure 2.a) was executed on the test data, with a misclassification rate of 0.2% and the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1062	2
p	2	1046

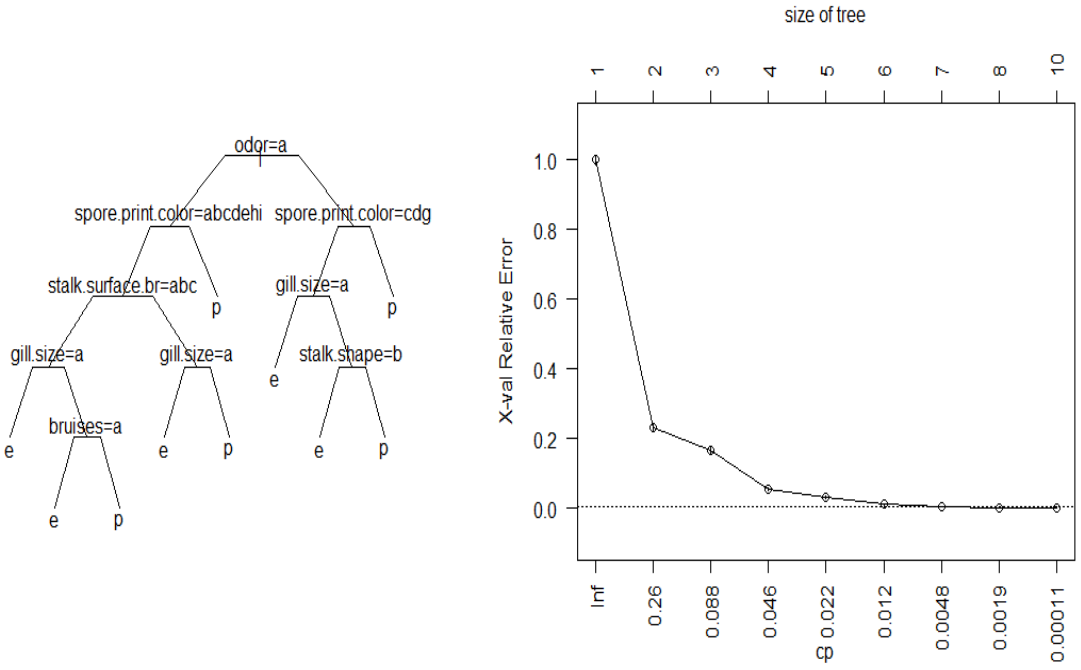

Figure 1: (a) Initial Classification Tree and (b) 10-fold Cross Validation Plot. Note that `rpart` has relabeled the qualitative factors.

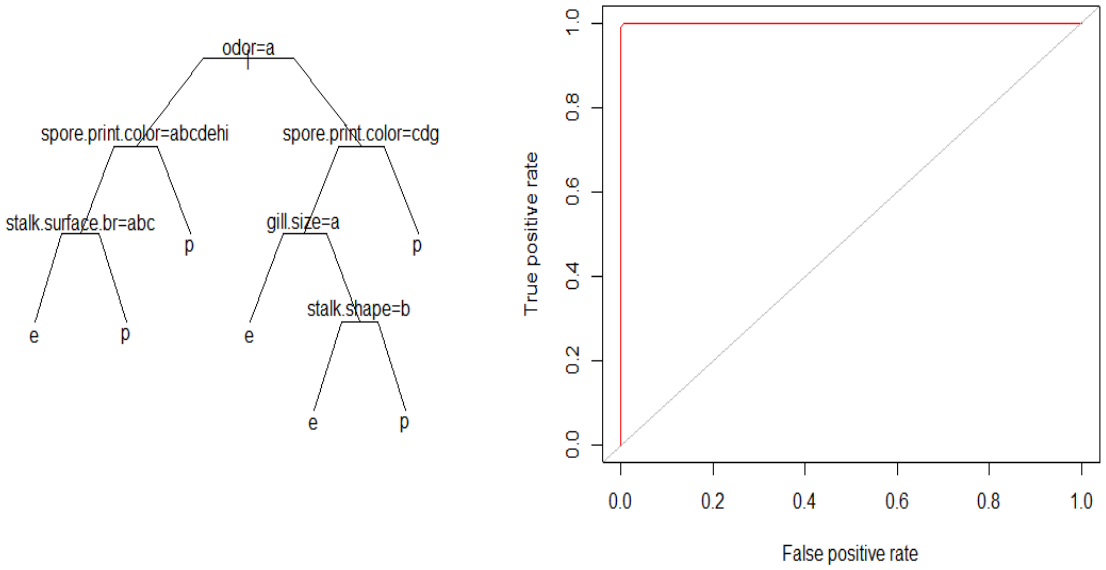

Figure 2: (a) 7 Node Classification Tree and (b) ROC for Positive Poisonous Prediction. Note that `rpart` has relabeled the qualitative factors.

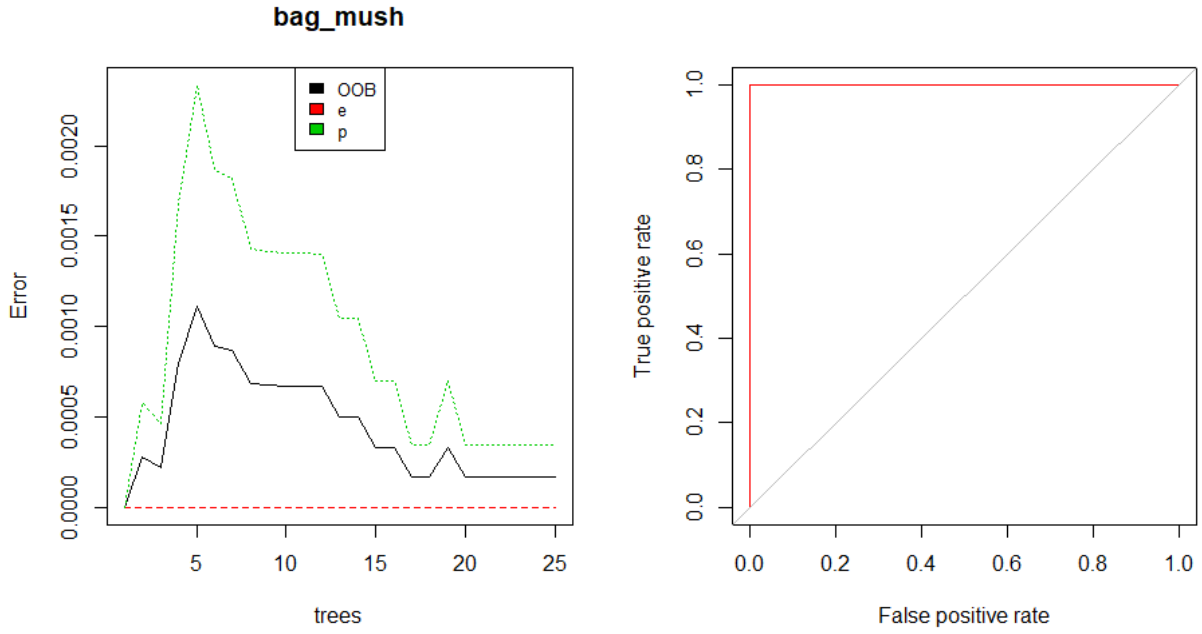

Figure 3: (a) Bagged Model Error per Number of Trees (b) ROC for Positive Poisonous Prediction

The ROC curve is shown in Figure 2.b with a AUC of 0.9992. The results shown an extreme level of correlation between the classifications and the predictors `odor`, `spore.print.color`, `gill.size`, `stalk.shape`, and `stalk.surface.br`. This high accuracy of the model is due to the well-separated nature of the data, which will be explored in later sections.

4.1.2 Random Forest

Two random forest type models were produced from the data set with the `stalk.root` feature removed: a bagged model with all features considered and a random forest model with only $\sqrt{20} \simeq 4$ features considered. The `randomForest` package and function was used to model both the bagged and random forest models.

The bagged model was first considered, and the results are shown in Figure 3. The OOB error reaches an equilibrium value around 25 trees. Predicting the test data set with 25 trees resulted in an AUC of 1.0 and the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1064	0
p	0	1048

This level of accuracy is expected given the precision of the single tree model in Section 4.1.1. The random forest with 4 features selected per tree is shown in Figure 4. The OOB error reaches an equilibrium value around 20 trees. Predicting the test data set with 20 trees resulted in an AUC of 1.0 and the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1064	0
p	0	1048

This level of accuracy is expected given the precision of the single tree model in Section 4.1.1. The OOB error converges more chaotically to the equilibrium value compared to the bagged model. The reason is due to the level of precision the 5 features identified in Section 4.1.1 have in classifying the mushrooms. Trees containing the 5 features perform far better at predicting the classifications than the models that do not, and since not all trees in the random forest consider these 5 features, the models converge slower to the equilibrium error.

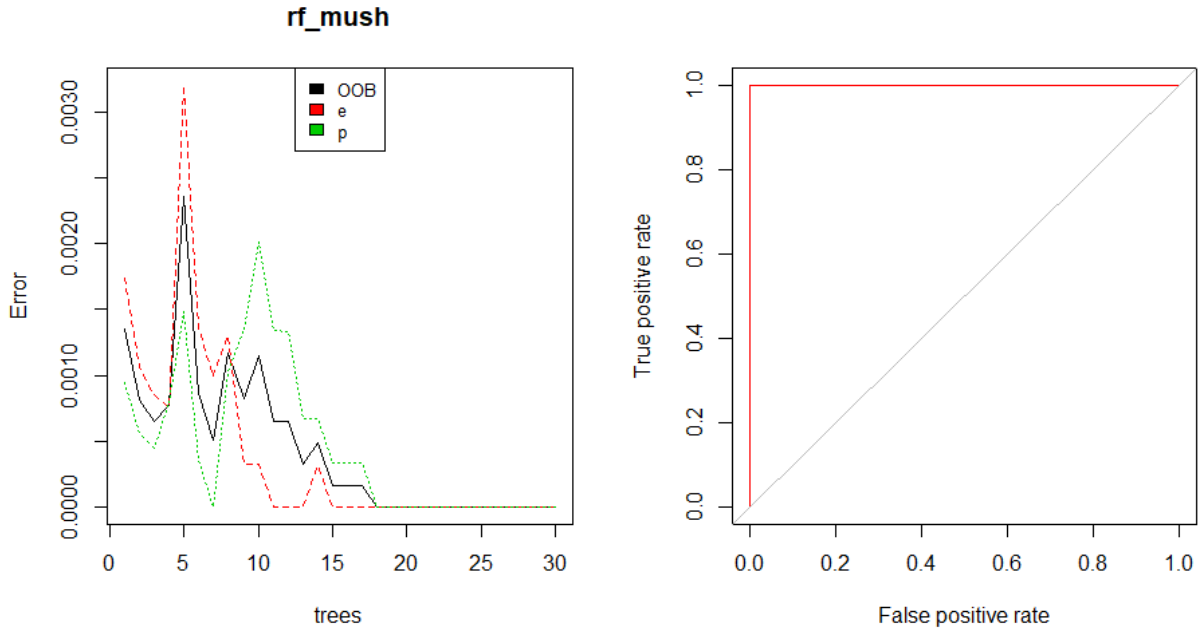

Figure 4: (a) Random Forest Model Error per Number of Trees (b) ROC for Positive Poisonous Prediction

4.1.3 Boosted Trees

A boosted tree model was produced from the full data set with the `stalk.root` feature removed. The `gbm` package and function was used. The `gbm` has two parameters to tune for model selection: the number of boosted trees and the interaction depth of each tree. The shrinkage parameter was maintained at the default `gbm` setting.

Figure 5 shows the 5-fold cross validation (CV) error as a function of the number of trees and the interaction depth. The CV error decreases quickly with the increase in interaction depth, eventually reaching an equilibrium value around 6-7. This result is consistent with the previous models, where the model will reach maximum prediction capability and additional predictors will provide little improvement. Figure 5 also shows the model continuously improving the CV error with the number of trees. The exponential trend of CV error shown in Figure 6 continues even to 5000 trees as the model successfully predicts marginally more values correctly.

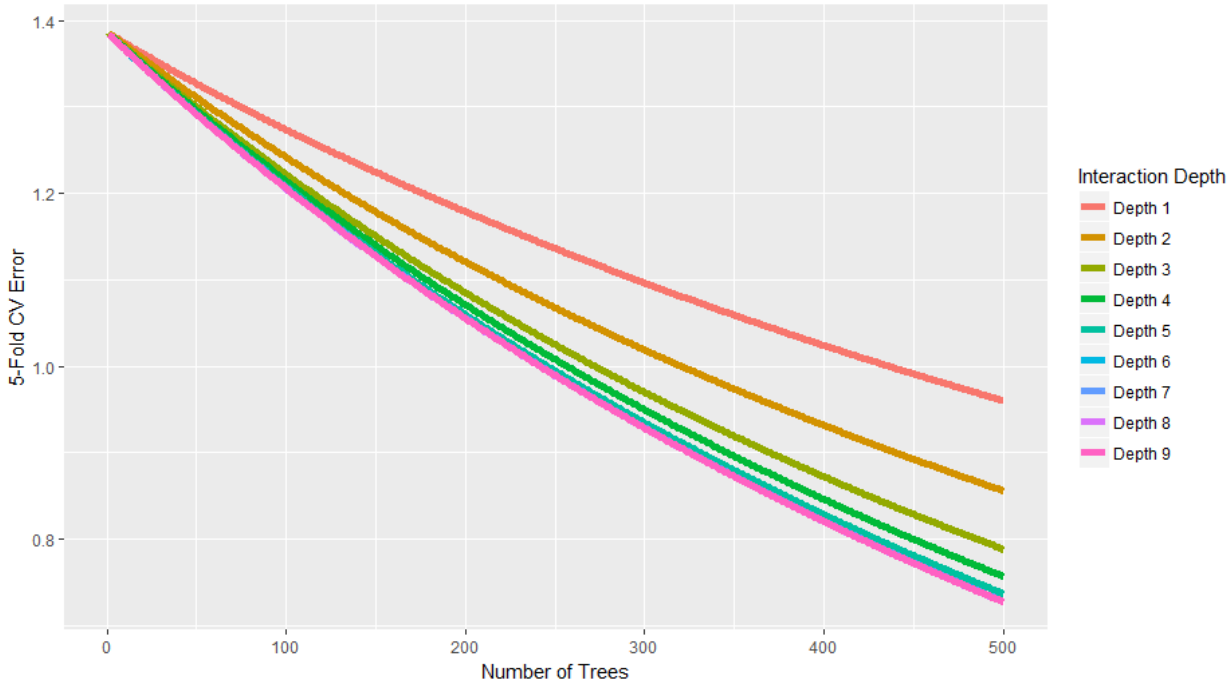

Figure 5: Interaction Depth Effect on Cross Validation Error

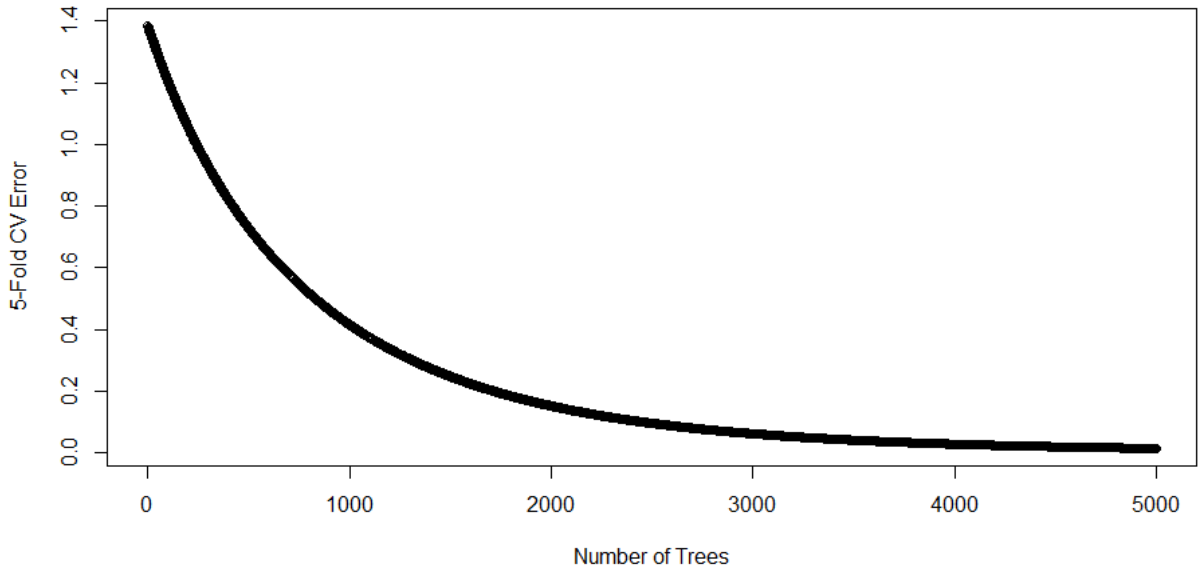

Figure 6: Cross Validation Error per Number of Trees (Interaction Depth 6)

The test set was predicted using the 5000 tree, 6 interaction depth boosted model with a 50% threshold value to produce the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1064	0
p	0	1048

The ROC AUC was calculated to be 1. The boosted model predicts the mushrooms with a high degree of accuracy. Previous executions with different test sets have shown that the boosted model can mis-predict non-conservatively due to misclassifying some poisonous mushrooms as edible. The boosted tree model is also substantially larger in size than the random forest type models for the same degree of accuracy. The boosted model uses 5000 trees to predict the test data with the same level of accuracy as the 20 tree random forest model, therefore the boosted model is not recommended.

4.2 Parametric Models

The parametric models for classification, Logistic, LDA, and QDA, were considered for modeling the mushroom toxicity classifications. For each of the models, only the predictors `odor`, `spore.print.color`, `gill.size`, `stalk.shape`, and `stalk.surface.br` identified in Section 4.1 were considered. This initial screening is justified because:
1. It is unlikely that additional data features will add substantial marginal predictive capability above what was found in Section 4.1, given the calculated variable importances.
2. A full model run with all 21 features would require 83 dummy variables to fully account for all qualitative features. Such a large model would be less parsimonious than the Section 4.1 models.
3. Multicollinearity was observed using the full feature set.

4.2.1 Logistic Model

The `glm` function was used to model the training data set with 5 predictors listed above. When executed, the `glm` produces the following warnings:

```
Warning messages:
1:  glm.fit:  algorithm did not converge
2:  glm.fit:  fitted probabilities numerically 0 or 1 occurred
```

These warnings have been shown [3] to arise in non-convergent GLM models due to complete separation. Complete separation will cause a non-convergence in the GLM model due to the likelihood function having no maximum [4, pp. 48]. Complete separation (or at least quasi-separation) is expected due to the extreme level of prediction seen in Section 4.1. To confirm the diagnosis, several `glm` models were executed without certain key predictors, and the model did converge in the non-optimum cases.

4.2.2 LDA Model

The `lda` function of the `MASS` package was used to fit an LDA model to the 5-predictor training data set. The test data misclassification error was 2.23% and produced the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1037	20
p	27	1028

The LDA underperformed all tree models in Section 4.1, however it does produce a solution because it doesn't require a convergence on the maximum likelihood function like the logistic model.

4.2.3 QDA Model

The `qda` function of the `MASS` package was used to fit a QDA model to the 5-predictor training data set. When executed, the model fails with the following error:

```
Error in qda.default(x, grouping, ...)  :  rank deficiency in group e
```

The diagnosis for this error is that the group variance could not be calculated due to a rank deficiency in inverting a covariance matrix. This is most likely due to the complete separation of data, which causes a variance of approximately 0 for both classification groups.

4.3 K-Nearest Neighbor Model

Similar to the Section 4.2 parametric models, a KNN model was also fit using the 5 highest value features determined in Section 4.1. The `knn` function of the `class` package was used to create the KNN model. Executing the `knn` function on the training data model matrix generates the following error:

```
Error in knn(train = train.X, test = test.X, cl = train.Y, k = 101) :
  too many ties in knn
```

This error stems from the use of many dummy variables in the model matrix taking on integer values of 0 or 1, which leads to too many matching cases when the `knn` function attempts to define the nearest neighborhood by Euclidean distance. To overcome this error, a random "jitter" was added to the model matrix data in the form of a random uniform number in the range [0.01, 0.05]. This jittering will allow the `knn` function to define a neighborhood within the [0,1] feature space without mixing the dummy variable responses.

The KNN model was run with 10-fold CV to determine k, the size of the neighborhood. Figure 7 shows the plot of 10-fold CV error in terms of the size of the neighborhood. The lowest CV error occurs when k = 1, or when the model selects the closest training value to define the prediction. This is an expected result based on previous sections, as the data is well separated and thus well defined by the training data. Averaging results from the training data, especially across the [0,1] dummy variable feature space, will lead to mispredictions.

The KNN model was run on the test data set with k = 1 and produced a misclassification rate of 0.09% with the following confusion matrix:

	Test Mushroom Data	
Predicted	e	p
e	1064	2
p	0	1046

The KNN model demonstrates a high level of prediction quality, however is does have a non-conservative misclassification bias.

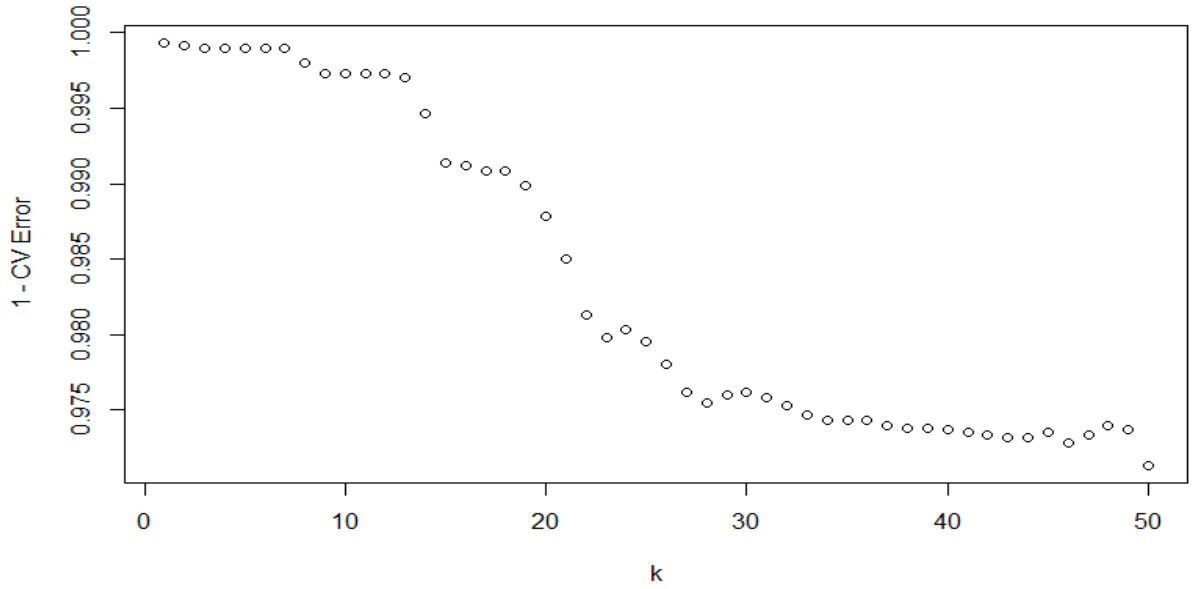

Figure 7: 10-fold Cross Validation Error per Neighborhood Size k

5 Conclusions

A number of the assessed models could not converge on a solution due to the well-separated nature of the data, however the models which do produce results show a high degree of accuracy. The classification trees in Section 4.1 produce results which both narrow down the required features to only 5 and give easily interpretable results. The bootstrapped tree ensembles further refine the accuracy of the model, eventually perfectly predicting the test data. The boosted model underperforms both the bagged and random forest models due to its large size, and should be avoided. The majority of the parametric models could not converge on a solution, due to the well-separated training data. The only parametric model that could be used was the LDA, which underperformed the tree models in accuracy and interpretablity. The KNN model was able to be executed after data jittering. KNN was able to work on par with the individual classification trees because the data is well predicted by the training data set, however it is less interpretable than the classification trees. It is concluded that the random forest classification model in Section 4.1.2 is the superior model, as it perfectly predicts the data, is easy to interpret, and uses fewer trees than the bagged tree model.

References

[1] The Audubon Society Field Guide to North American Mushrooms (1981). G. H. Lincoff (Pres.), New York: Alfred A. Knopf

[2] Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository, `https://archive.ics.uci.edu/ml/datasets/Mushroom`. Irvine, CA: University of California, School of Information and Computer Science.

[3] "Logistic regression model does not converge." StackExchange, `https://stats.stackexchange.com/questions/5354/logistic-regression-model-does-not-converge`. Posted December 20, 2010. Retrieved April 22, 2018.

[4] Vector Generalized Linear and Additive Models: With an Implementation in R (2015). T. W. Yee, New York: Springer

A Appendix: Mushroom.csv Summary

```
 class    cap.shape cap.surface   cap.color    bruises       odor
 e:4208   b: 452    f:2320      n      :2284   f:4748   n      :3528
 p:3916   c:   4    g:   4      g      :1840   t:3376   f      :2160
          f:3152    s:2556      e      :1500            s      : 576
          k: 828    y:3244      y      :1072            y      : 576
          s:  32                w      :1040            a      : 400
          x:3656                b      : 168            l      : 400
                                (Other): 220            (Other): 484
 gill.attachment gill.spacing gill.size   gill.color   stalk.shape stalk.root
 a: 210          c:6812       b:5612    b      :1728   e:3516      b   :3776
 f:7914          w:1312       n:2512    p      :1492   t:4608      c   : 556
                                        w      :1202               e   :1120
                                        n      :1048               r   : 192
                                        g      : 752               NA's:2480
                                        h      : 732
                                        (Other):1170
 stalk.surface.above.ring stalk.surface.below.ring stalk.color.above.ring
 f: 552                   f: 600                   w      :4464
 k:2372                   k:2304                   p      :1872
 s:5176                   s:4936                   g      : 576
 y:  24                   y: 284                   n      : 448
                                                   b      : 432
                                                   o      : 192
                                                   (Other): 140
 stalk.color.below.ring veil.type veil.color ring.number ring.type
 w      :4384           p:8124    n:  96     n:  36      e:2776
 p      :1872                     o:  96     o:7488      f:  48
 g      : 576                     w:7924     t: 600      l:1296
 n      : 512                     y:   8                 n:  36
 b      : 432                                            p:3968
 o      : 192
 (Other): 156
 spore.print.color population habitat
 w      :2388      a: 384     d:3148
 n      :1968      c: 340     g:2148
 k      :1872      n: 400     l: 832
 h      :1632      s:1248     m: 292
 r      :  72      v:4040     p:1144
 b      :  48      y:1712     u: 368
 (Other): 144                 w: 192
```