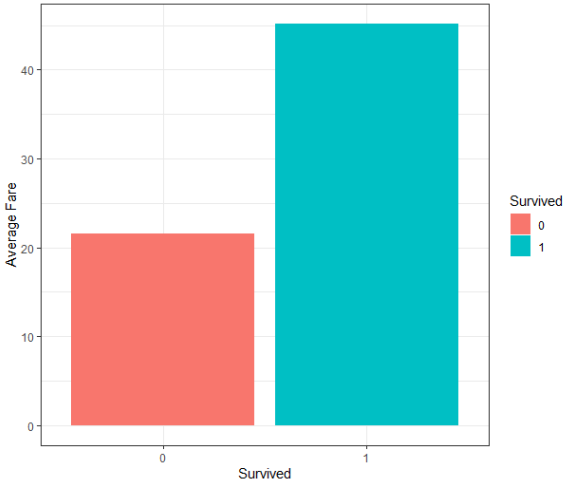
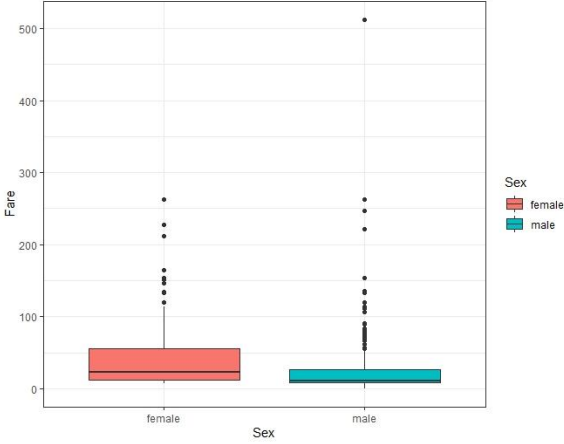
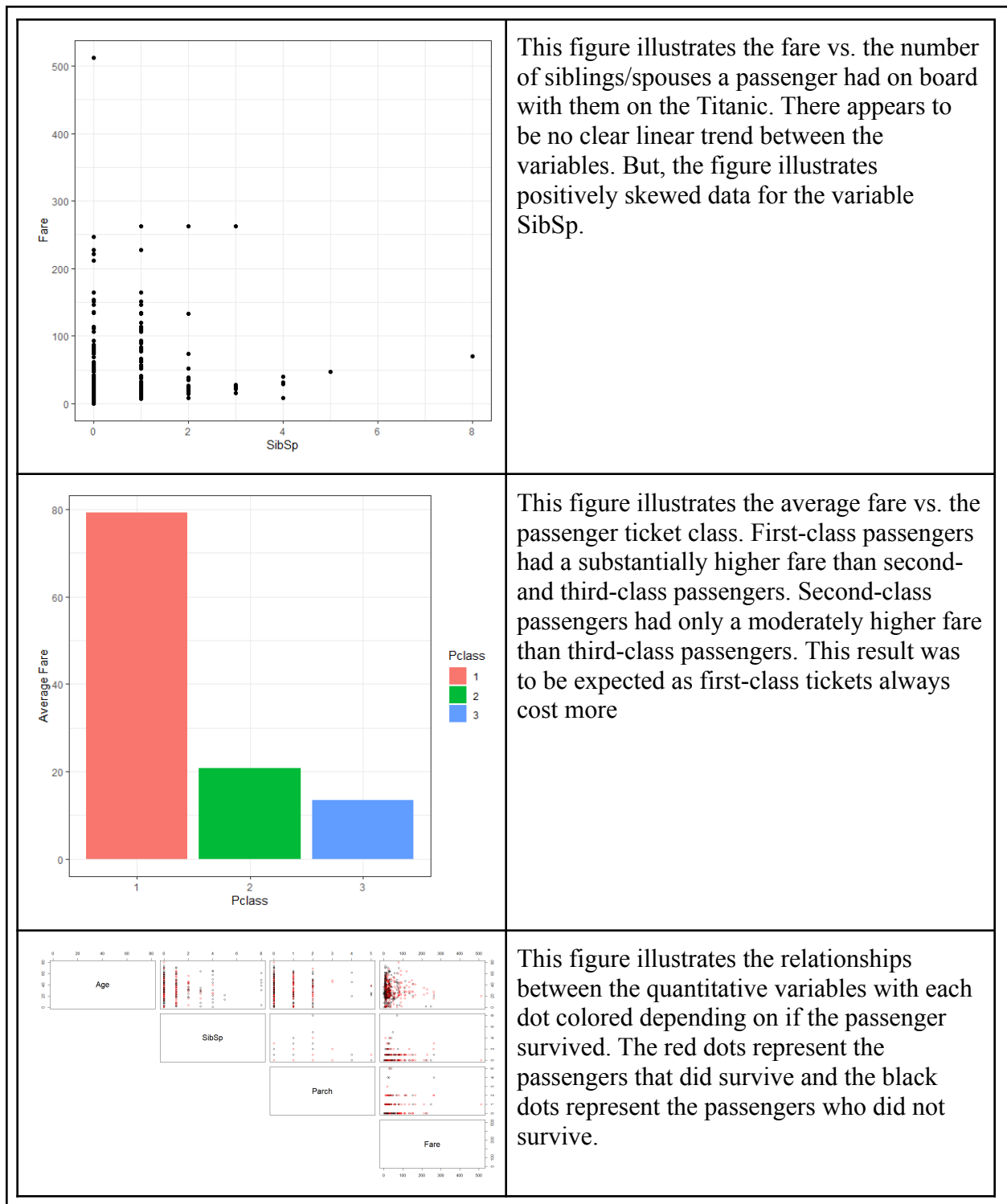


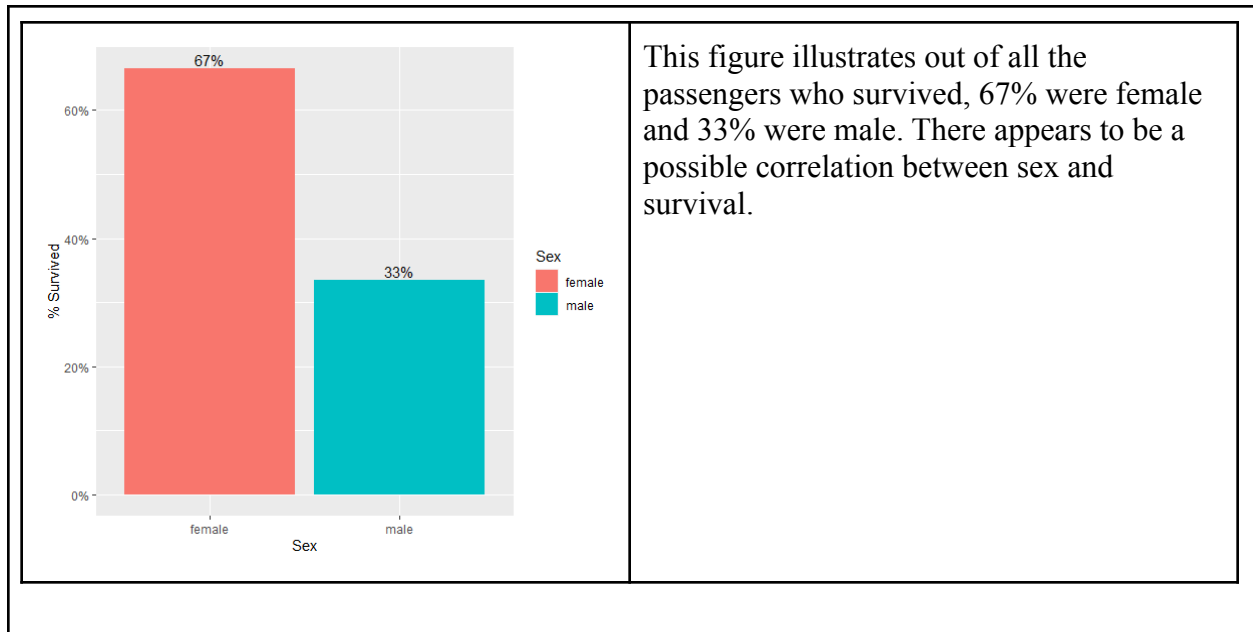
**Background:** Our regression problem is to determine if we can predict the passenger fare using a variety of covariates. Our classification problem is to determine if we can predict whether a passenger survived the Titanic's sinking using a variety of covariates. The variables for port of embarkation and passenger class were converted into factors with three levels each, and the variables for sex and survival were converted into factors with two levels each. The data frame of 891 observations was split 70-30 into a training set (623 observations) and a testing set (268 observations), and there were no missing values.

**Regression Problem:** Four methods were chosen to approach the regression problem: a naive method, OLS, ridge, and lasso regression. For the naive method, the sample mean of the fares for the training set's passengers - 30.93614 pounds - was used as the prediction for the fares of the test set's passengers. For ridge regression and lasso regression, the tuning parameter was chosen via 10-fold cross validation. We found that the optimal lambda for ridge regression was 1.929175, and the optimal lambda for lasso was 0.06617977 (Appendix F). Interestingly, at the optimal lambda for lasso, no variable selection was performed as there were still nine covariates in the model. Appendices D and E present the outputs of OLS, ridge, and lasso regression. From it, we can see that both ridge and lasso regression's coefficients were shrunk towards zero, but ridge regression's shrinkage was generally more severe. The MSPEs for all four methods are shown in Appendix B. Overall, when the models were applied to testing data, the naive sample mean method (MSPE = 3580.933) was determined to have the worst performance while OLS (MSPE = 2249.013) had the best performance. It's likely that OLS performed better than both shrinkage methods because the increase in bias was greater than the decrease in variance. Lasso (MSPE = 2251.337) performed better than ridge regression (MSPE = 2277.664), and this might be because lasso's less severe coefficient shrinkage allowed it to be less biased and more closely resemble OLS's model than ridge regression. The signs of the coefficients of all three regression models inform us that survival and traveling with more family members were associated with more expensive fares while embarking from Southampton and Queenstown, being older, being male, and having a lower class ticket were associated with a less expensive fare.

**Classification Problem:** For our classification question we decided that it would be best to use logistic regression given that we have a mix of categorical and quantitative input variables. Also, logistic regression is superior to methods such as linear discriminant analysis because it does not require the usual linear regression assumptions such as normality and homoscedasticity. After fitting the model, however, we still needed to check that multicollinearity is not going to affect our model, which is one of the few assumptions of logistic regression. To check this we simply checked the variance inflation factors for all of our input variables and all the VIF coefficients were below 2, which means we can be pretty certain that multicollinearity is not present (Appendix G). After fitting our model and checking the VIF coefficients, we tested our model on test data. Using our 267 test observations we received an accuracy rate of roughly 74.25%. This was lower than all three of the other methods that we looked at such as naive bayes, LDA and QDA (Appendix C). The FPR is 14.89% and the FNR is 31.61%. It is no surprise that our FNR is higher than FPR given the unbalance in our data. Overall, logistic regression underwhelmed us in its prediction ability as it had the lowest accuracy.

Appendix A: Exploratory Data Analysis	
Figure	Description
	<p>This figure illustrates the average fare for survivors vs. non-survivors. Survivors paid, on average, more than double the fare of non-survivors. There appears to be a positive association between survival and fare.</p>
	<p>This figure illustrates the fare for females vs. males. Females had a higher median fare. Furthermore, the fare is much more spread out among females compared to males. In general, it appears females paid a higher fare than males. However, the figure indicates the distribution of the fare is positively skewed for both sexes.</p>





#### Appendix B: Comparison of Regression Methods MSPE

Naive	OLS	Ridge	Lasso
3580.933	2249.013	2277.664	2251.337

#### Appendix C: Comparison of Classification Methods Accuracy

Naive Bayes	Logistic Regression	LDA	QDA
0.7873134	0.7425373	0.7835821	0.7947761

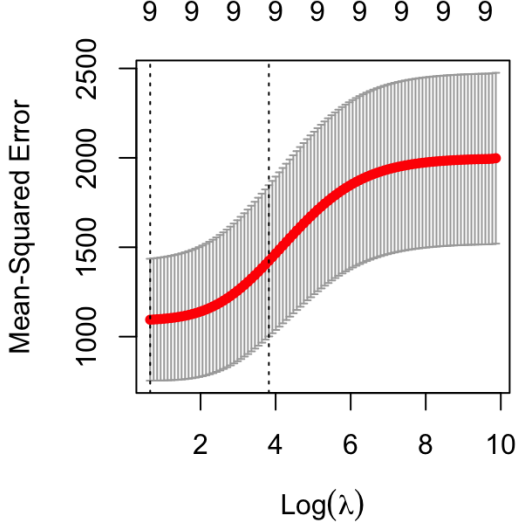
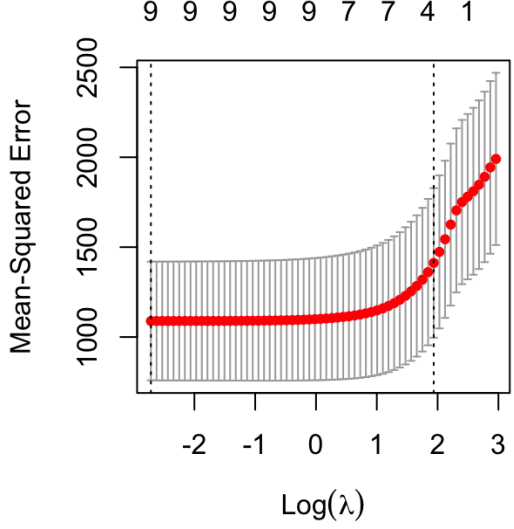
#### Appendix D: Output of OLS Regression

```
> coef(ols)
10 x 1 sparse Matrix of class "dgCMatix"
      s0
(Intercept) 84.17065181
Survived1    2.42799654
Pclass2     -55.67538242
Pclass3     -63.82541180
Sexmale      -4.85521247
Age          -0.09276794
SibSp        5.91449883
Parch        9.78408209
EmbarkedQ    -8.71219024
EmbarkedS   -10.56070301
```

#### Appendix E: Output of Ridge and Lasso Regression

<pre>&gt; coef(ridge) 10 x 1 sparse Matrix of class "dgCMatix"       s0 (Intercept) 79.74395258 Survived1    3.73376898 Pclass2     -49.59379815 Pclass3     -57.83536302 Sexmale      -4.70008288 Age          -0.08825074 SibSp        5.66840616 Parch        9.33682209 EmbarkedQ    -10.36478845 EmbarkedS   -11.16773614</pre>	<pre>&gt; coef(lasso) 10 x 1 sparse Matrix of class "dgCMatix"       s0 (Intercept) 83.63459779 Survived1    2.41821872 Pclass2     -55.34919744 Pclass3     -63.60080242 Sexmale      -4.77519873 Age          -0.08785925 SibSp        5.86926270 Parch        9.73838705 EmbarkedQ    -8.36146707 EmbarkedS   -10.34147756</pre>
--	---

## Appendix F: Cross-Validation for Ridge and Lasso Regression

Ridge	Lasso
	
<pre>&gt; lambda.ridge [1] 1.929175</pre>	<pre>&gt; lambda.lasso [1] 0.06617977</pre>

## Appendix G: VIF Coefficients

	GVIF	Df	GVIF^(1/(2*Df))
Sex	1.254601	1	1.120090
Pclass	1.837180	2	1.164228
SibSp	1.259110	1	1.122101
Embarked	1.232959	2	1.053749
Parch	1.341439	1	1.158205
Fare	1.691259	1	1.300484
Age	1.030148	1	1.014962

## Appendix H: Logistic Regression Output

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	2.763758	0.475662	5.810	6.23e-09	***
Sexmale	-2.762089	0.242728	-11.379	< 2e-16	***
Pclass2	-0.453617	0.339489	-1.336	0.18149	
Pclass3	-1.492784	0.325893	-4.581	4.64e-06	***
SibSp	-0.207360	0.111829	-1.854	0.06370	.
EmbarkedQ	-0.628973	0.451793	-1.392	0.16387	
EmbarkedS	-0.817231	0.276475	-2.956	0.00312	**
Parch	-0.096304	0.139925	-0.688	0.49129	
Fare	0.002637	0.003260	0.809	0.41863	
Age	0.004619	0.007453	0.620	0.53541	

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 837.59 on 622 degrees of freedom  
 Residual deviance: 571.49 on 613 degrees of freedom  
 AIC: 591.49

**Regression Problem:** We began by creating a regression tree. Cross-validation found the optimal cost parameter to be 0. When the decision tree was applied to test data, the MSPE was found to be 2201.03 (Appendix B). The decision tree is shown in Appendix C. Because the optimal cost parameter was 0, no pruning was performed. Our random forest model ( $mtry = 2$ ) and bagging ( $mtry = 7$ ) performed similarly, with MSPEs of 2100.803 (Appendix B). Finally, we applied gradient boosting to our regression problem. The optimal number of trees was found to be 7221 through 10-fold cross-validation (Appendix I), and interaction depth was set to 1. The MSPE for boosting was found to be 2299.528 (Appendix B). Of the four methods, boosting surprisingly performed the worst while random forests/bagging performed the best. Boosting's poor performance may be because a single split per tree might be too simple to explain the data properly, even with 7221 trees built sequentially. Random forests/bagging performed better than the regression tree, and this may be because the regression tree overfit the training data and because random forests/bagging allowed better tree splits to occur that would not have been possible in the regular decision tree due to greedy splitting. Appendices F, G, and J show the variable importance plots, and Appendix E shows the decision tree. Interestingly, all methods agreed that passenger class was the single most important variable in predicting fare, and the number of parents and children ( $parch$ ) was a distant second in most models.

**Classification Problem:** Our first nonlinear method is the basic decision tree. After training the model and using cross validation to find the best complexity parameter, our final classification model had a complexity parameter of 0.022. After the best complexity parameter and retraining the model, our misclassification rate using the test data was 19.03%. Our second nonlinear method is random forest. The tuning parameter for random forest is the argument  $mtry$ , which specifies the number of predictors that are candidates for each split. Typically, the value for this tuning parameter is the square root of the number of predictors, rounded down. Since there are seven predictors, the best value for  $mtry$  is two. After performing random forest with this value for the tuning parameter, our misclassification rate using the test data is 18.28%. Our third nonlinear method is bagging. The tuning parameter for bagging is the same as random forest,  $mtry$ . However, for bagging, the value for this tuning parameter is the number of predictors, which is seven. After performing bagging with this value for the tuning parameter, our misclassification rate using the test data is 21.27%. Our final nonlinear method is boosting. The tuning parameter is  $n.trees$ , the number of trees to build. We used 10-fold cross-validation to choose the best number of trees to build. After performing boosting with the optimal number of trees, our misclassification rate using the test data is 21.64%. Therefore, random forests performed the best.

**Comparison of Performance Between Nonlinear and Linear Methods:** Appendices C and D compare nonlinear methods with our linear methods. We can see that, overall, our tree based models performed better. Our best performing model was our random forest model. Our worst performing model was our logistic regression model. Between the best and worst models, there is some slight overlap, with bagging obtaining the same error rate as naive bayes and boosting obtaining the same error rate as LDA. The reason why our linear methods struggle with predicting may have to do with the lack of linearity of our response variable. Linear boundaries such as what can be found in LDA and logistic regression, do not perform as well whereas non linear decision boundaries such as decision trees and QDA perform better because they may be better able to accurately pick up on nonlinearities in the feature space.



**Appendix A: Comparison of Misclassification Rate (Classification)**

<b>Decision Tree</b>	<b>Random Forest</b>	<b>Bagging</b>	<b>Boosting</b>
0.1902985	0.1828358	0.2126866	0.2164179

**Appendix B: Comparison of MSPE (Regression)**

<b>Decision Tree</b>	<b>Random Forest</b>	<b>Bagging</b>	<b>Boosting</b>
2201.0300	2100.8030	2100.8030	2299.528

**Appendix C: Comparison of Nonlinear and Linear Methods' Misclassification Rate (Classification)**

<b>Decision Tree</b>	<b>Random Forest</b>	<b>Bagging</b>	<b>Boosting</b>
0.1902985	0.1828358	0.2126866	0.2164179

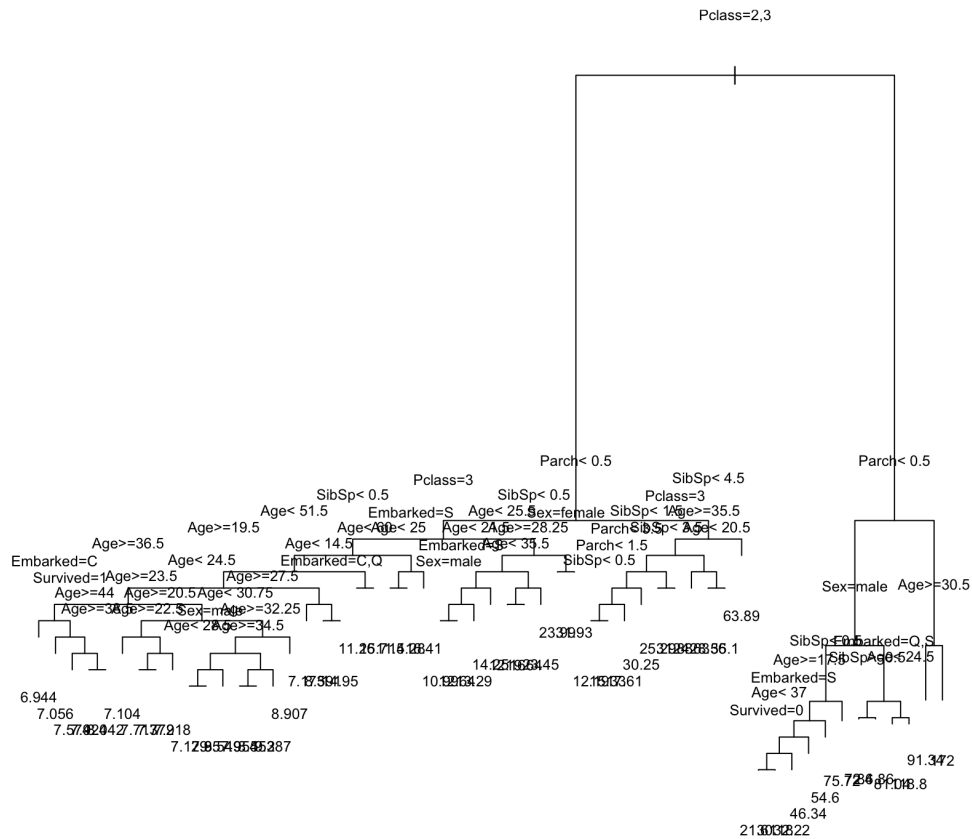
<b>Naive Bayes</b>	<b>Logistic Regression</b>	<b>LDA</b>	<b>QDA</b>
0.2126866	0.2574627	0.2164179	0.2052239

**Appendix D: Comparison of Nonlinear and Linear Methods' MSPE (Regression)**

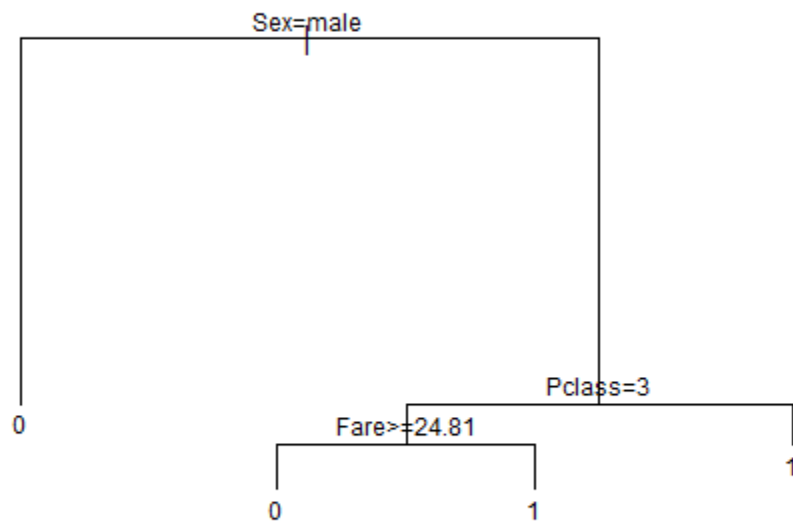
<b>Decision Tree</b>	<b>Random Forest</b>	<b>Bagging</b>	<b>Boosting</b>
2201.0300	2100.8030	2100.8030	2299.528

<b>Naive</b>	<b>OLS</b>	<b>Ridge</b>	<b>Lasso</b>
3580.933	2249.013	2277.664	2251.337

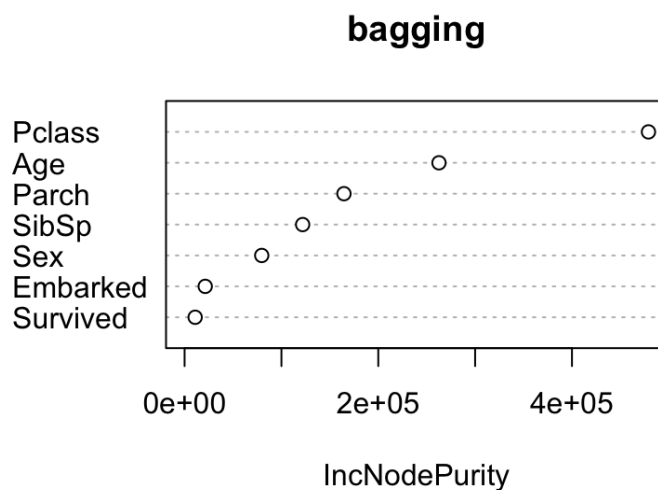
## Appendix E: Decision Tree with Complexity Parameter = 0.0 (Regression)



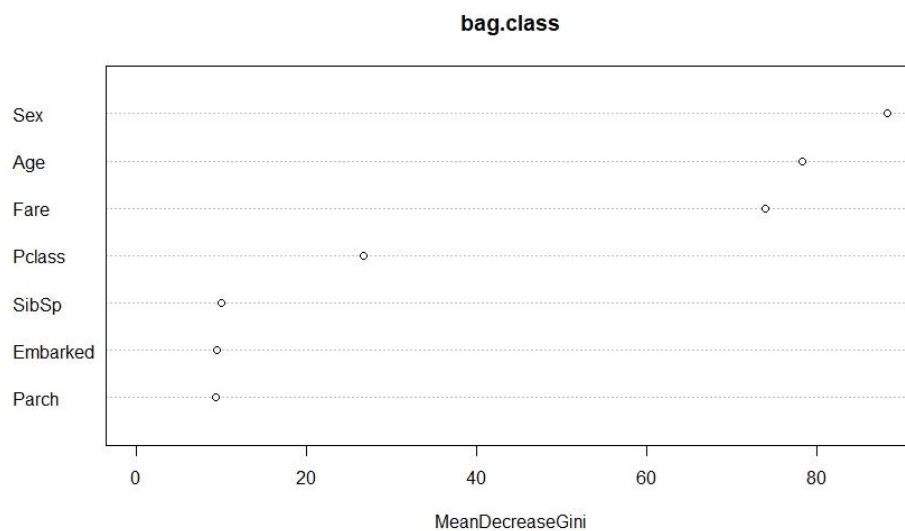
## Appendix F: Decision Tree with Complexity Parameter = 0.022 (Classification)



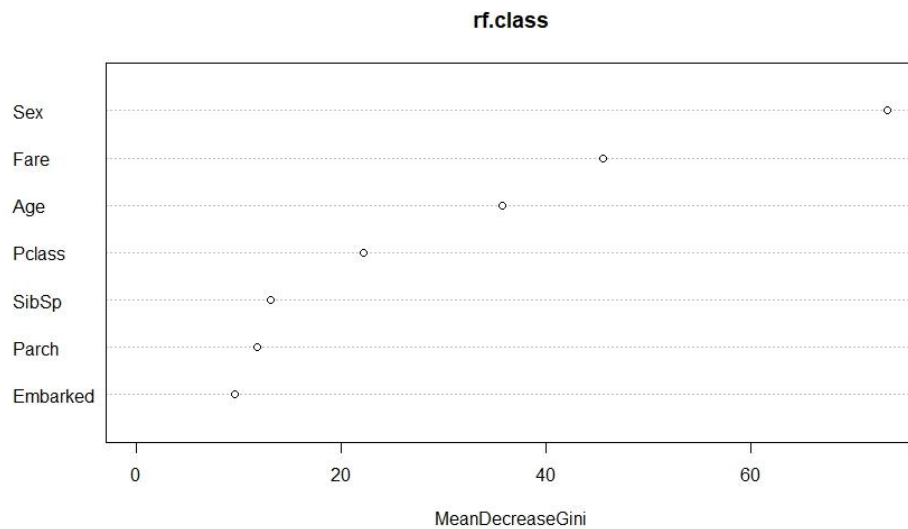
### Appendix G: Bagging Variable Importance Plot for Regression Problem



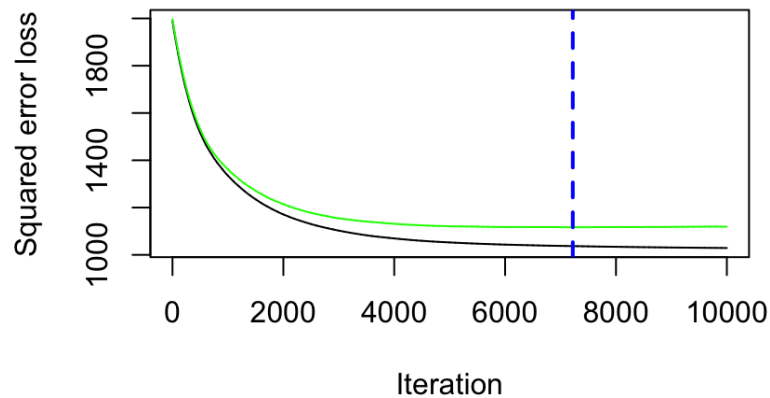
### Appendix H: Bagging Variable Importance Plot for Classification Problem



## Appendix H: Random Forest Importance Plot for Classification Problem



## Appendix I: 10-fold Cross Validation for Boosting to Select the Optimal Number of Trees (Regression)

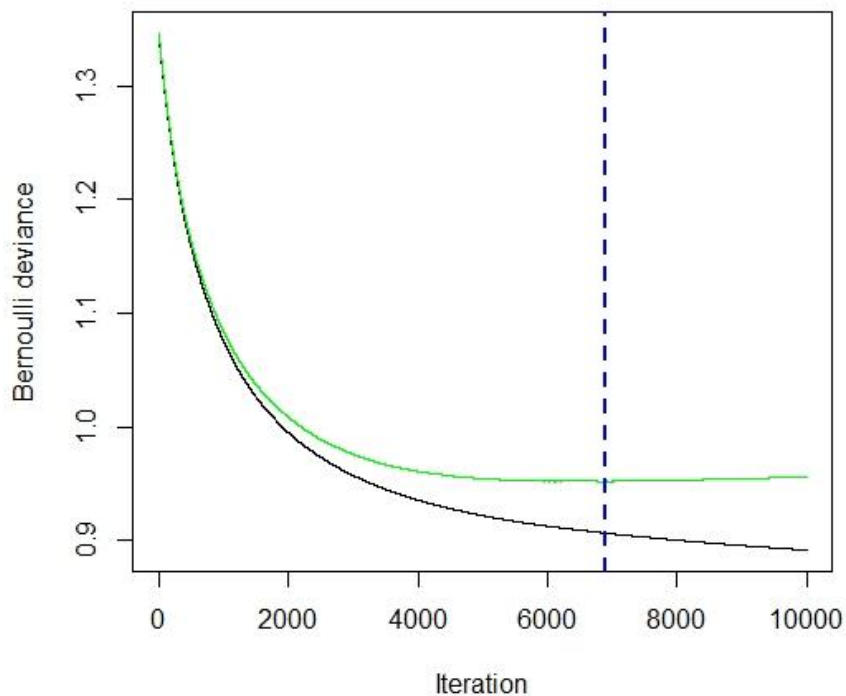


A gradient boosted model with gaussian loss function.  
10000 iterations were performed.  
The best cross-validation iteration was 7221.  
There were 7 predictors of which 7 had non-zero influence.

**Appendix J:** Variable Importance for Boosting (Regression)

	var	rel.inf
Pclass	Pclass	71.8107541
Parch	Parch	14.4676809
SibSp	SibSp	6.6545295
Age	Age	2.8911491
Sex	Sex	2.0452823
Embarked	Embarked	1.5557048
Survived	Survived	0.5748991

**Appendix K:** 10-fold Cross Validation for Boosting to Select the Optimal Number of Trees (Classification)



A gradient boosted model with bernoulli loss function.  
10000 iterations were performed.  
The best cross-validation iteration was 6890.  
There were 7 predictors of which 7 had non-zero influence.

Appendix L: Variable Importance for Boosting (Classification)		
		rel.inf
Sex	57.163351	
Pclass	15.902001	
Fare	9.274288	
Age	6.612239	
Embarked	4.917296	
SibSp	4.872435	
Parch	1.258390	