INFO 6105
Data Science Engineering Methods and Tools

Northeastern University, Spring 2019

Problem Set 2, Due: Dec 10, 2019

Problem Set Rules:

1. Each student should hand in an individual problem set.
2. Discussing problem sets with other students is permitted. Copying from another person or solution set is *not* permitted.
3. Late assignments will *not* be accepted. No exceptions.
4. The solutions should be posted on the website by the beginning of class on the day the problem set is due.
5. How to get data sets: See Package ISLR

1. (Total: 20 points) We want to use logistic regression to predict the probability of default using income, balance, and student on the Default data set. Our goal is to estimate the test error of this logistic regression model using the validation set approach. Do not forget to set a random seed before beginning your analysis.
   (a) (5 points) Using the validation set approach, estimate the test error of a logistic regression model that uses income and balance to predict default. In order to do this, you must perform the following steps:
      1. Split the sample set into a training set and a validation set.
      2. Fit a multiple logistic regression model using only the training observations.
      3. Obtain a prediction of default status for each individual in the validation set by computing the posterior probability of default for that individual, and classifying the individual to the default category if the posterior probability is greater than 0.5.
      4. Compute the validation set error, which is the fraction of the observations in the validation set that are misclassified.

```
#1. Split the sample set into a training set and a validation set.
```

```
X = df[['income', 'balance']]
y = df['default_yes']
X_train, X_test, y_train, y_test = train_test_split(df, y, test_size=0.5)
```

```
#2. Fit a multiple logistic regression model using only the training observations.
```

```
lr = LogisticRegression(C=10**6, tol=1e-6)
mod = lr.fit(X_train, y_train)
```

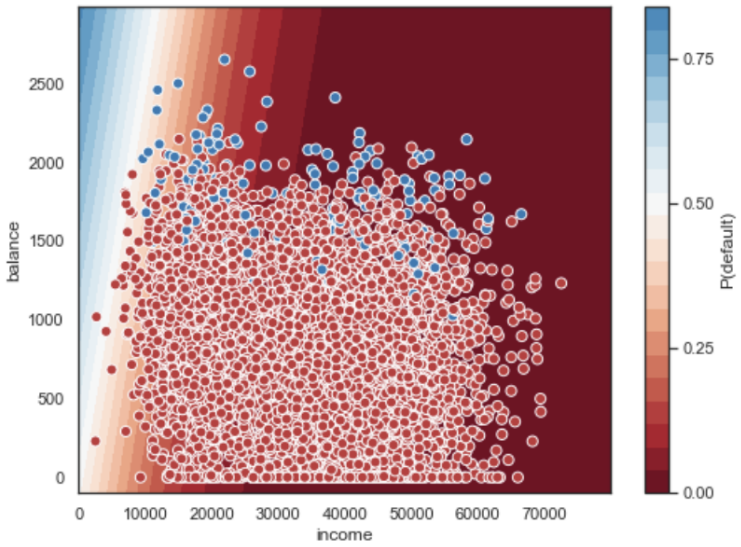
```
#3. Obtain a prediction of default status for each individual in the validation set
```

```
xx, yy = np.mgrid[0:80000:100, -100:3000:10]
grid = np.c_[xx.ravel(), yy.ravel()]                    # https://www.quora.com/Can-anybody-elaborate-the-use-of
probs = mod.predict_proba(grid)[:, 1].reshape(xx.shape)
```

```
f, ax = plt.subplots(figsize=(8,6))
contour = ax.contourf(xx, yy, probs, 25, cmap="RdBu",    # 25 levels
                      vmin=0, vmax=1)
ax_c = f.colorbar(contour)
ax_c.set_label("P(default)")
ax_c.set_ticks([0,0.25,0.5,.75,1])

ax.scatter(X_test['income'], X_test['balance'], c=y_test, s=50,
           cmap="RdBu", vmin=-0.2, vmax=1.2,
           edgecolor="white", linewidth=1)

ax.set(xlabel="income", ylabel="balance");
```




```
#4. Compute the validation set error
```

```
y_pred = mod.predict(X_test)
1-(y_pred == y_test).mean()
```

(b) (7 points) Repeat the process in (a) three times, using three different splits of the observations into a training set and a validation set. Describe your findings and comment on the results obtained.

```
y_pred = mod.predict(X_test)
1-(y_pred == y_test).mean()
```

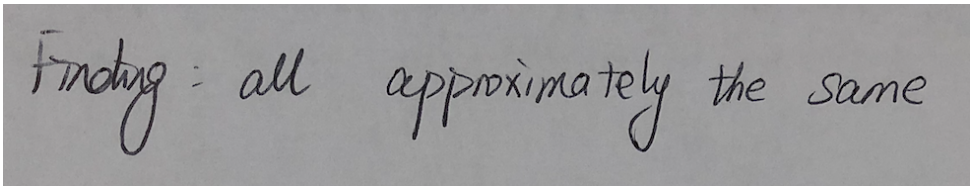
```
0.032200000000000006
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4)
y_pred = mod.predict(X_test)
1-(y_pred == y_test).mean()
```

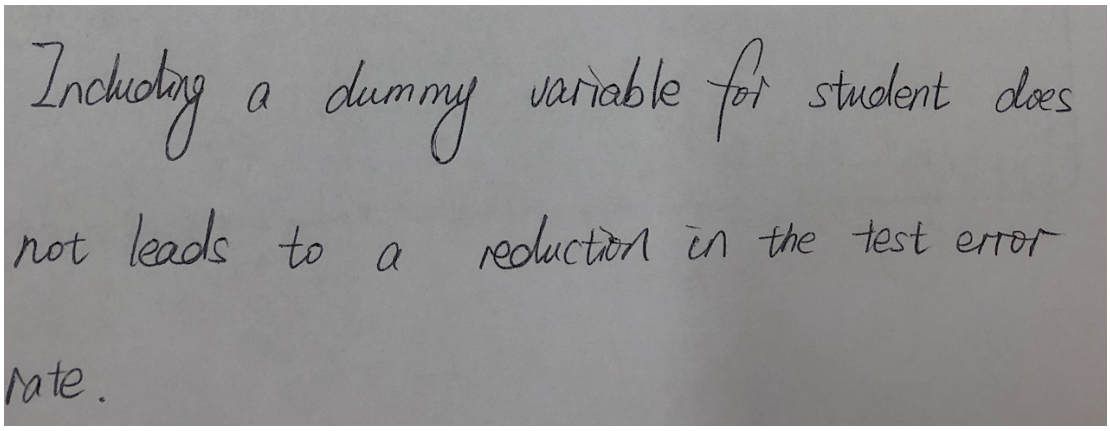
```
0.03374999999999995
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.6)
y_pred = mod.predict(X_test)
1-(y_pred == y_test).mean()
```

```
0.03466666666666662
```



(c) (8 points) Now consider a logistic regression model that predicts the probability of default using income, balance, and a dummy variable for student. Estimate the test error for this model using the validation set approach. Comment on whether or not including a dummy variable for student leads to a reduction in the test error rate.



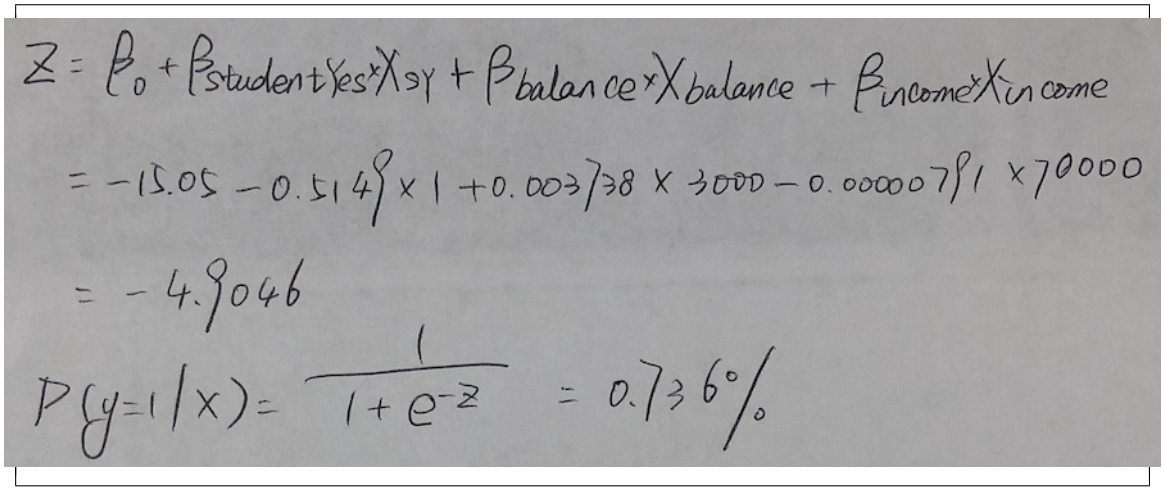
2. (Total: 20 points)

Suppose we collect data for a group of bank customers with variables

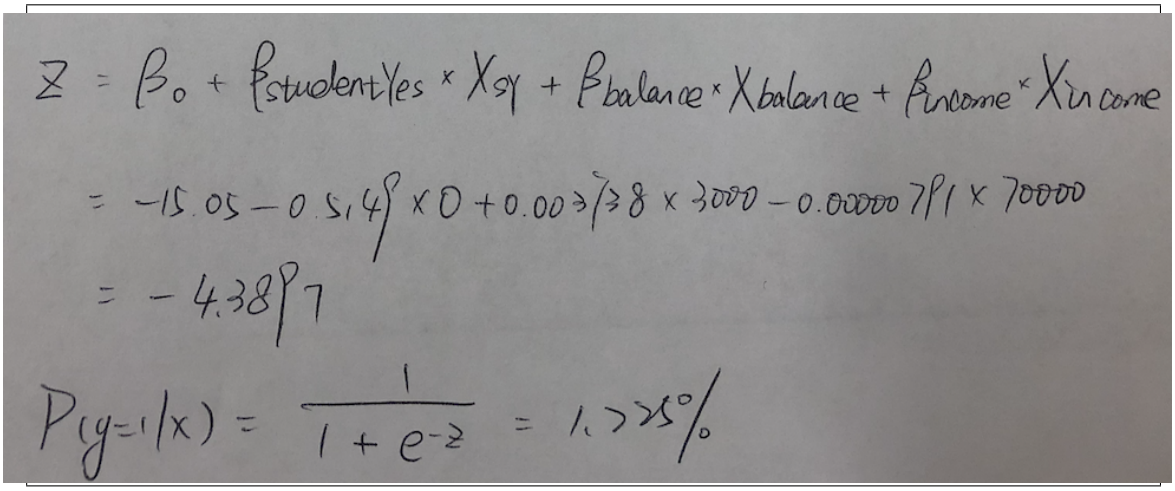
- default A variable with levels No and Yes indicating whether the customer defaulted on their debt
- student A variable with levels No and Yes indicating whether the customer is a student
- balance The average balance that the customer has remaining on their credit card after making their monthly payment
- income Income of customer

We fit a logistic regression and produce estimated coefficient, $\hat{\beta}_0 = -15.05$, $\hat{\beta}_{\text{studentYes}} = -0.5149$, $\hat{\beta}_{\text{balance}} = 0.003738$, $\hat{\beta}_{\text{income}} = -0.00000791$.

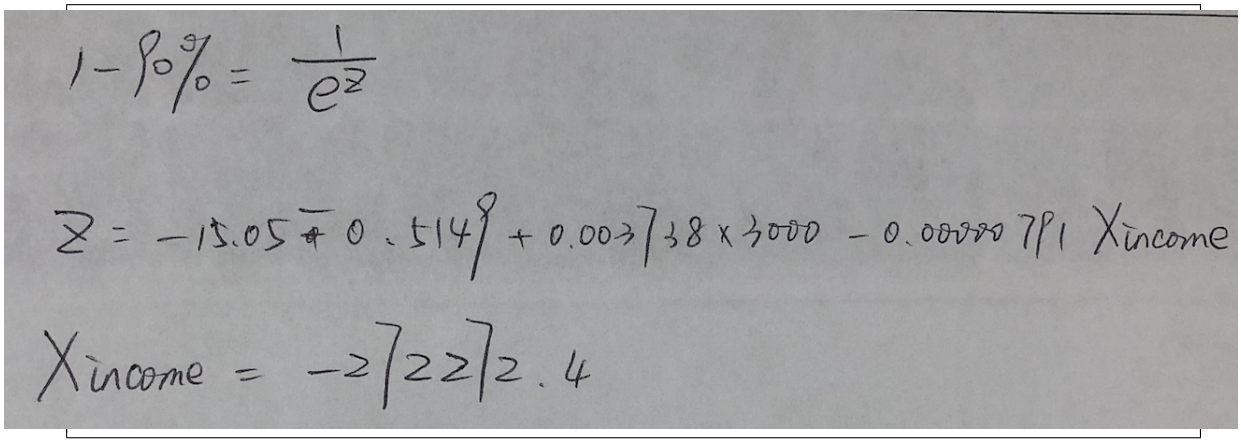
(a) (5 points) Estimate the probability that a student with a balance of \$3,000 and income \$70,000 does default on a loan.



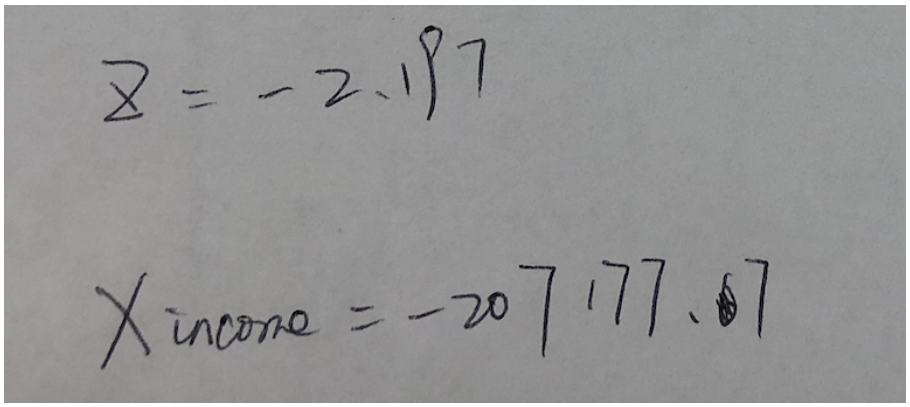
(b) (5 points) Estimate the probability that borrower a balance of \$3,000 and income \$70,000 who is not a student does default on a loan.



(c) (5 points) How much income would the student in part (a) need to make to have a 90% chance of getting approved for a loan?



(d) (5 points) How much income would the borrower in part (b) need to make to have a 90% chance of getting approved for a loan?



3. (Total: 30 points) This problem involves the Boston data set, which can be found in the file Boston.csv. This data set contains the following columns:

crim per capita crime rate by town.

zn proportion of residential land zoned for lots over 25,000 sq.ft.

indus proportion of non-retail business acres per town.

chas Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).

nox nitrogen oxides concentration (parts per 10 million).

rm average number of rooms per dwelling.

age proportion of owner-occupied units built prior to 1940.

dis weighted mean of distances to five Boston employment centres.

rad index of accessibility to radial highways.

tax full-value property-tax rate per $10,000.

ptratio pupil-teacher ratio by town.

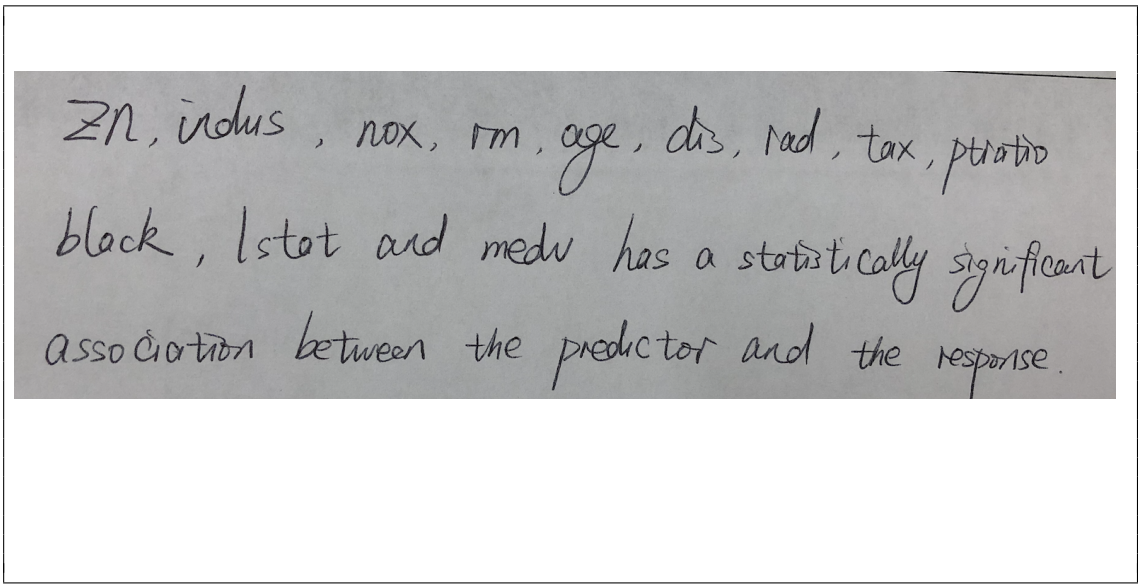
black $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.

lstat lower status of the population (percent).

medv median value of owner-occupied homes in $1,000s.

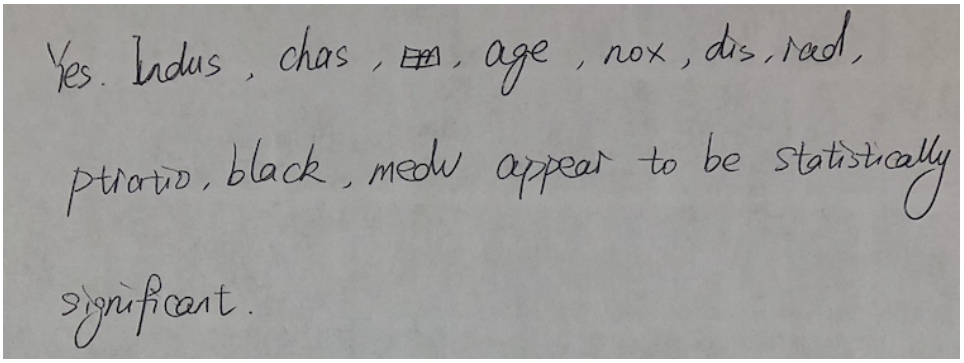
We want to predict whether a given suburb has a crime rate above or below the median using the other variables in this data set.

(a) (5 points) For each predictor, fit a simple logistic regression model to predict the response. In which of the models is there a statistically significant association between the predictor and the response (set $\alpha = 0.05$)

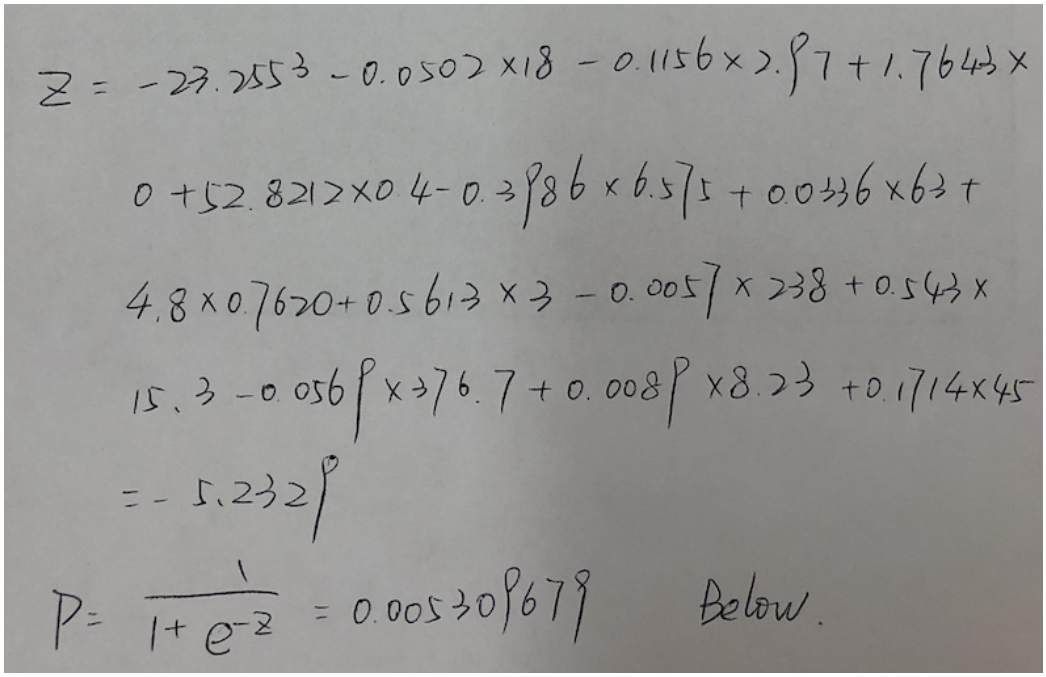


(b) (12 points) Fit a multiple logistic model to predict the response using all of the predictors. Describe your results.

(i) Do any of the predictors appear to be statistically significant (set $\alpha = 0.05$)? If so, which ones?

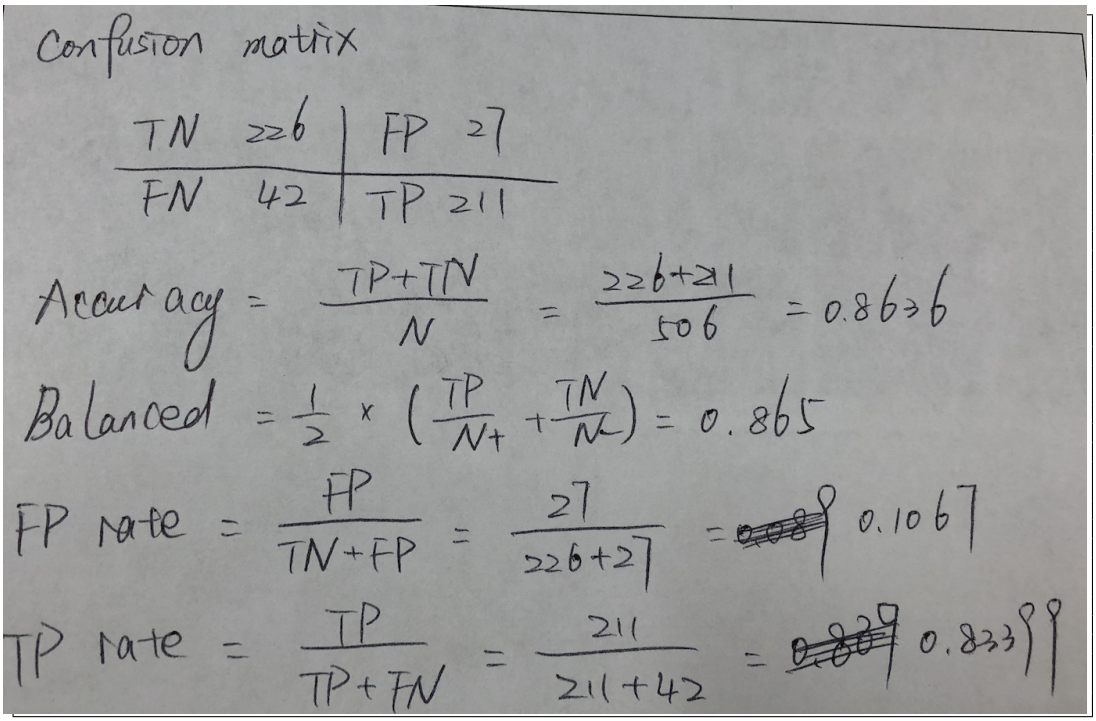


(ii) Predict whether a given suburb has a crime rate above or below the median for a suburb with:
zn=18 indus=2.97 chas= 0 nox= 0.4 rm=6.575 age=63 dis=4.8
rad=3 tax=238 ptratio=15.3 black=376.7 lstat=8.23 medv=45

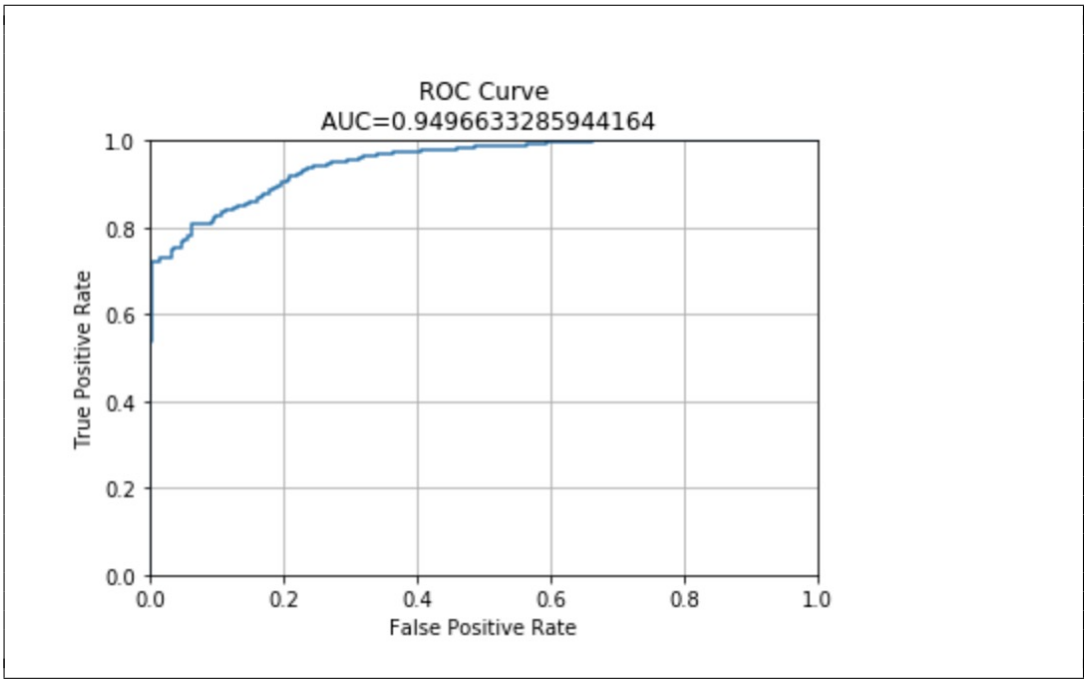


(iii) Compute the confusion matrix, accuracy, balanced accuracy, FP, and FP rates for Threshold=0.5. Explain what the confusion matrix is telling you about the types of

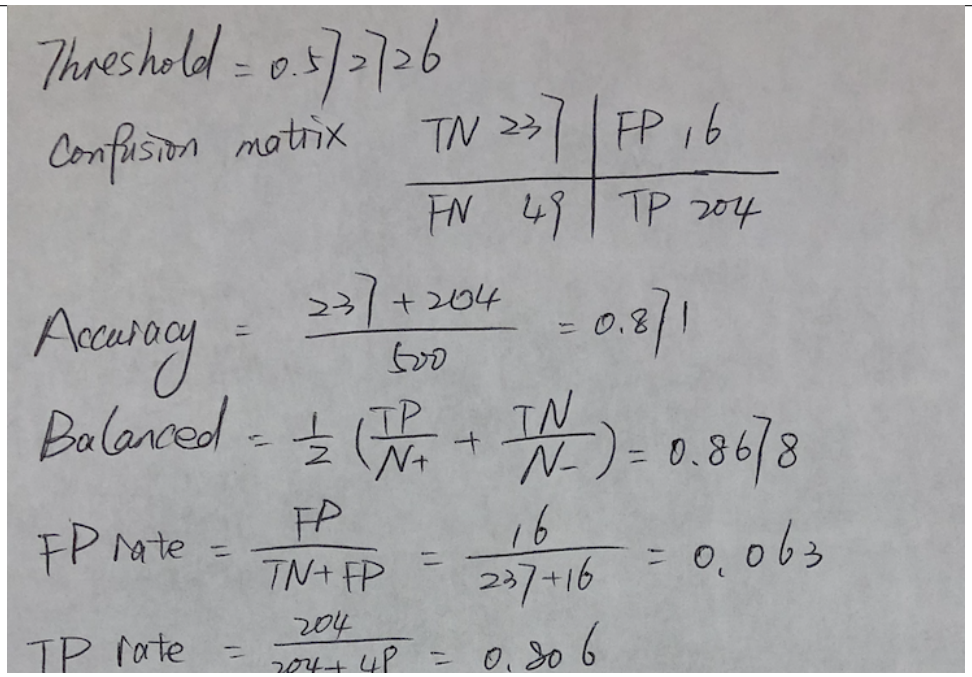
mistakes made by logistic regression.



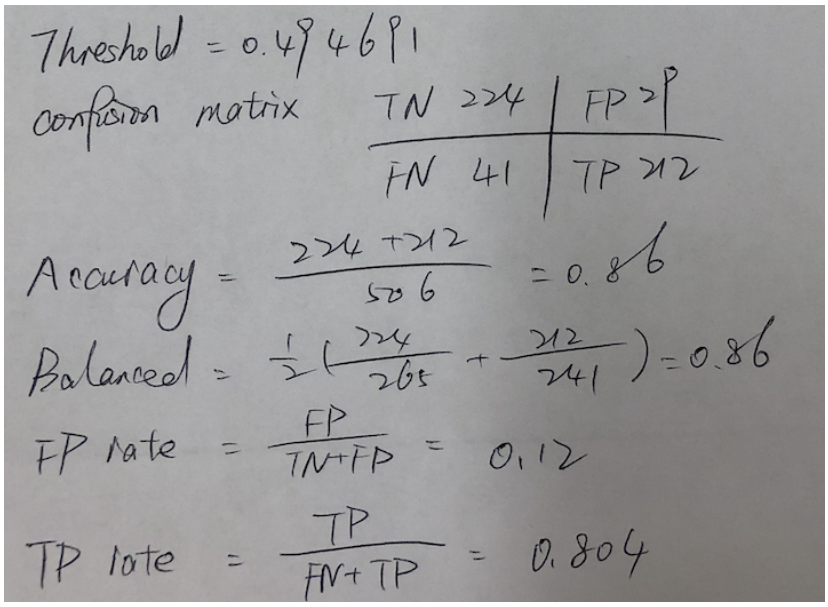
(iv) Draw the ROC curve. What is the AUC?



(v) Determine the threshold for which (TP rate + (1-FP rate)) is maximal. Then, compute the confusion matrix, accuracy, balanced accuracy, FP, and FP rates for that threshold.



(vi) Determine the threshold for which the ROC curve has the minimum distance to the upper left corner (where TP rate=1 and FP rate=0). Note that this distance is $\sqrt{(1 - \text{TP rate})^2 + (\text{FP rate})^2}$. Then, compute the confusion matrix, accuracy, balanced accuracy, FP, and FP rates for that threshold.



(c) (6 points) Which predictors matter most for predicting predicting whether a given suburb has a crime rate above or below the median? (Find the first and the second most important variables)

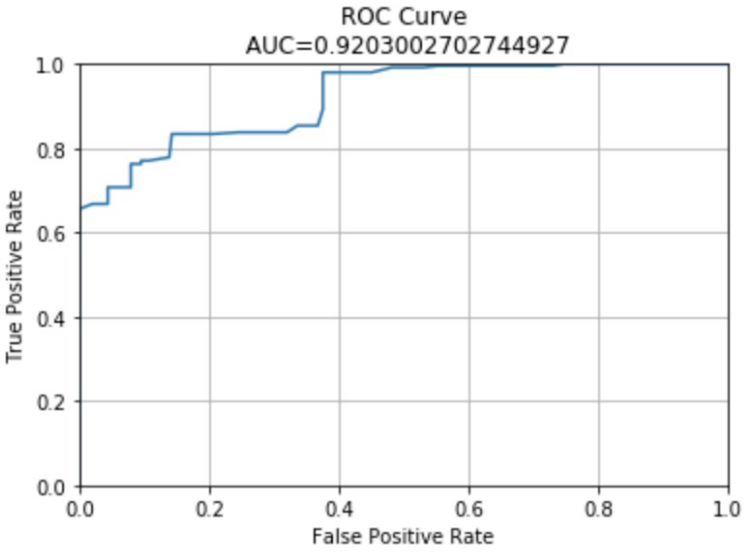
nox and rad is the first and second most important variables.

(d) (7 points) Fit a multiple logistic model to predict the response using the first two most important variables in Part (c).

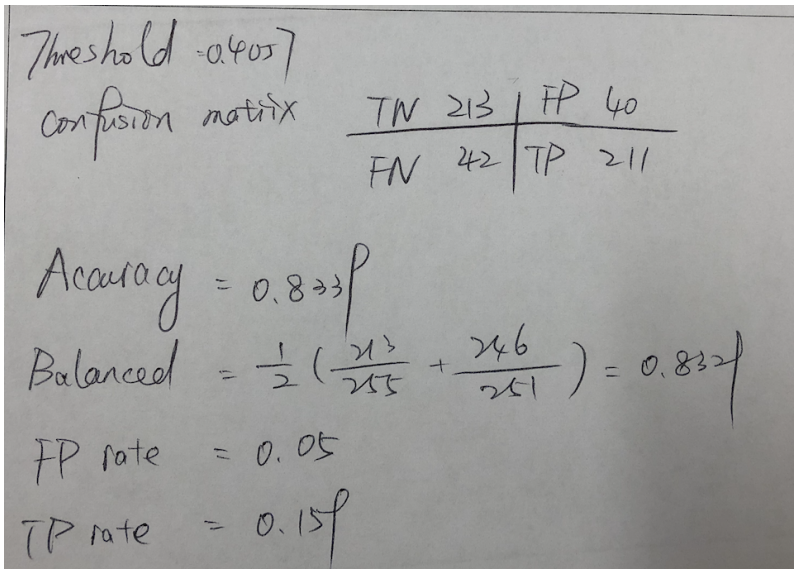
(i) Write out the model in equation form.

$$P(y=1|x) = \frac{1}{1+e^{-(-17.7362+28.0206(nox)+0.5073(rad))}}$$

(ii) Draw the ROC curve. What is the AUC?



(iii) Determine the threshold for which (TP rate + (1-FP rate)) is maximal. Then, compute the confusion matrix, accuracy, balanced accuracy, FP, and FP rates for that threshold.

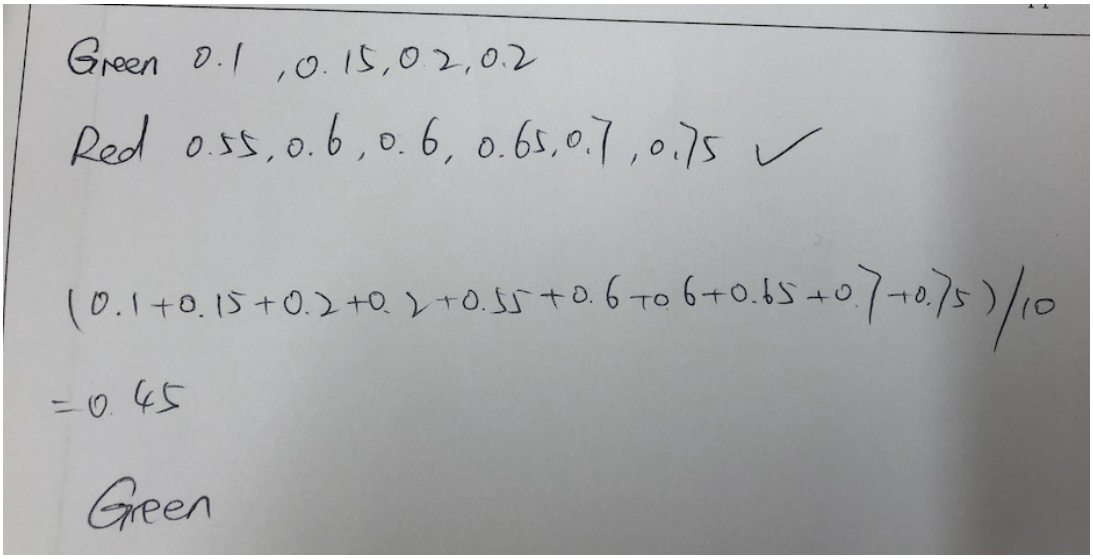


4. (Total: 8 points) Suppose we produce ten bootstrapped samples from a data set containing red and green classes. We then apply a classification tree to each bootstrapped sample and,

for a specific value of X, produce 10 estimates of P(Class is Red | X):

$$0.1, 0.15, 0.2, 0.2, 0.55, 0.6, 0.6, 0.65, 0.7, \text{ and } 0.75.$$

There are two common ways to combine these results together into a single class prediction. One is the majority vote approach. The second approach is to classify based on the average probability. In this example, what is the final classification under each of these two approaches?



5. (Total: 26 points) This problem involves the Carseats data set. We want to predict Sales using regression trees. Split the data set into a training set (75%) and a test set (0.25%).

   (a) (6 points) Fit a regression tree to the training set. Plot the tree, and interpret the results. What test MSE do you obtain?

```
library(ISLR)
set.seed(666)
train <- sample(1:nrow(Carseats), nrow(Carseats) * 0.75)
Carseats.train <- Carseats[train, ]
Carseats.test <- Carseats[-train, ]
```

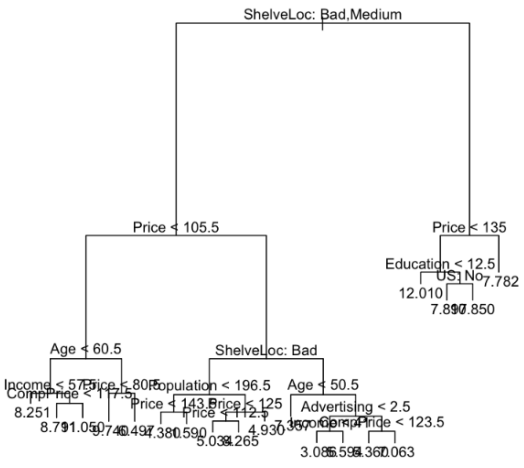
```
library(tree)
tree.carseats <- tree(Sales ~ ., data = Carseats.train)
summary(tree.carseats)
```

```
Regression tree:
tree(formula = Sales ~ ., data = Carseats.train)
Variables actually used in tree construction:
[1] "ShelveLoc"   "Price"       "Age"         "Income"      "CompPrice"
[6] "Population"  "Advertising" "Education"   "US"
Number of terminal nodes:  19
Residual mean deviance:  2.416 = 679 / 281
Distribution of residuals:
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-4.68700 -0.98320 -0.01679  0.00000  1.05900  3.83800
```

```
yhat <- predict(tree.carseats, newdata = Carseats.test)
mean((yhat - Carseats.test$Sales)^2)
```

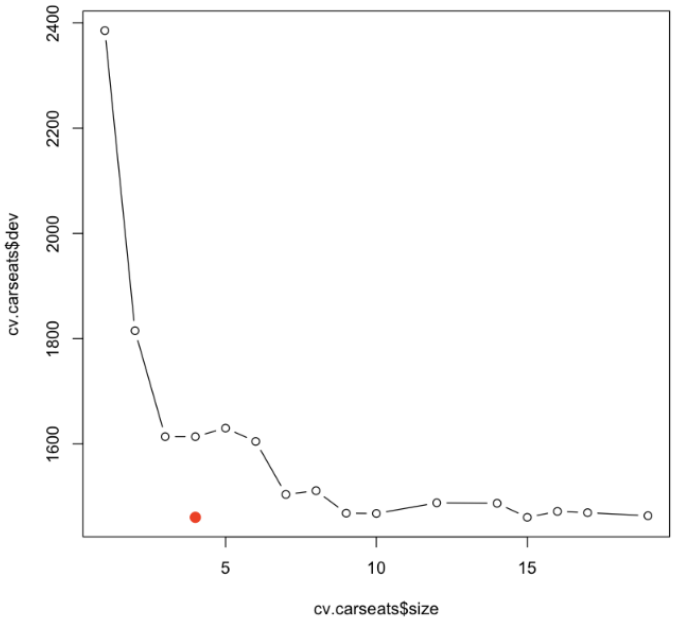
```
5.21796104698601
```

```
plot(tree.carseats)
text(tree.carseats, pretty = 0)
```

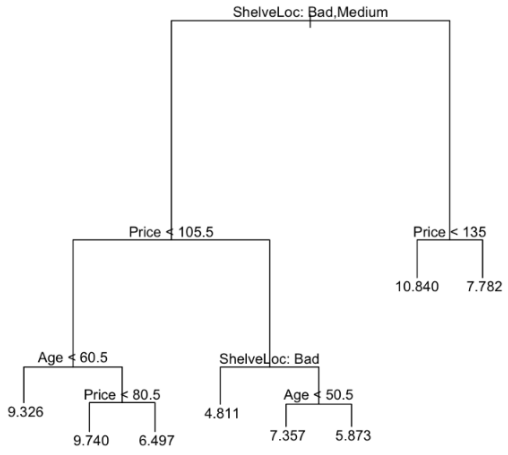


   (b) (6 points) Use cross-validation in order to determine the optimal level of tree complexity. Does pruning the tree improve the test MSE?

```
cv.carseats <- cv.tree(tree.carseats)
plot(cv.carseats$size, cv.carseats$dev, type = "b")
tree.min <- which.min(cv.carseats$dev)
points(tree.min, cv.carseats$dev[tree.min], col = "red", cex = 2, pch = 20)
```




```
prune.carseats <- prune.tree(tree.carseats, best = 8)
plot(prune.carseats)
text(prune.carseats, pretty = 0)
```



(c) (7 points) Use the bagging approach in order to analyze this data. What test MSE do you obtain? Determine which variables are most important.

```
bag.carseats <- randomForest(Sales ~ ., data = Carseats.train, mtry = 10, ntree = 500, importance = TRUE)
yhat.bag <- predict(bag.carseats, newdata = Carseats.test)
mean((yhat.bag - Carseats.test$Sales)^2)
```

2.75122112555638

```
importance(bag.carseats)
```

A matrix: 10 × 2 of type dbl

	%IncMSE	IncNodePurity
CompPrice	31.0708444	241.56034
Income	12.1118850	139.89585
Advertising	23.4965576	175.22847
Population	-0.9892401	82.95186
Price	67.3375566	697.00514
ShelveLoc	66.7128162	651.03724
Age	24.5283336	204.11548
Education	0.3603859	59.29979
Urban	-3.3623140	10.02579
US	5.4845076	17.60845

(d) (7 points) Use random forests to analyze this data. What test MSE do you obtain? Determine which variables are most important. Describe the effect of the number of variables considered at each split on the error rate obtained.

```
rf.carseats <- randomForest(Sales ~ ., data = Carseats.train, mtry = 3, ntree = 500, importance = TRUE)
yhat.rf <- predict(rf.carseats, newdata = Carseats.test)
mean((yhat.rf - Carseats.test$Sales)^2)
```

3.05621610347915

```
importance(rf.carseats)
```

A matrix: 10 × 2 of type dbl

	%IncMSE	IncNodePurity
CompPrice	14.1966217	195.92287
Income	5.3735784	179.07413
Advertising	18.5203059	227.70977
Population	0.1147289	149.39535
Price	43.8766699	559.86096
ShelveLoc	48.7813074	511.54548
Age	15.2692919	236.56267
Education	2.0614942	90.03332
Urban	-1.6571524	17.80725
US	3.5360887	30.06433

6. (Total: 26 points) This question uses the Caravan data set. Create a training set consisting of the first 1,000 observations, and a test set consisting of the remaining observations.

(a) (10 points) Fit a boosting model to the training set with Purchase as the response and the other variables as predictors. Use 1,000 trees, and a shrinkage value of 0.01. Which predictors appear to be the most important?

```
library(ISLR)
library(gbm)
library(MASS)

set.seed(6)
train <- 1:1000
Caravan$Purchase <- ifelse(Caravan$Purchase == "Yes", 1, 0)
Caravan.train <- Caravan[train, ]
Caravan.test <- Caravan[-train, ]
```

```
In [3]: summary(boost.caravan)
```

A data.frame: 85 × 2

	var	rel.inf
	<fct>	<dbl>
PPERSAUT	PPERSAUT	13.5323309
MKOOPKLA	MKOOPKLA	11.2820861
MOPLHOOG	MOPLHOOG	6.8274615
MBERMIDD	MBERMIDD	5.5332019
PBRAND	PBRAND	4.8598648
ABRAND	ABRAND	4.3433708
MGODGE	MGODGE	4.3318631
MINK3045	MINK3045	3.8878807
PWAPART	PWAPART	2.6115989
MAUT2	MAUT2	2.5725196

(b) (16 points) Use the boosting model to predict the response on the test data. Predict that a person will make a purchase if the estimated probability of purchase is greater than 20 %. Form a confusion matrix. What fraction of the people predicted to make a purchase do in fact make one? How does this compare with the results obtained from applying logistic regression to this data set?

```
probs.test <- predict(boost.caravan, Caravan.test, n.trees = 1000, type = "response")
pred.test <- ifelse(probs.test > 0.2, 1, 0)
table(Caravan.test$Purchase, pred.test)
```

```
   pred.test
       0    1
  0 4498   35
  1  279   10
```

```
logit.caravan <- glm(Purchase ~ ., data = Caravan.train, family = "binomial")
```

```
Warning message:
“glm.fit: fitted probabilities numerically 0 or 1 occurred”
```

```
probs.test2 <- predict(logit.caravan, Caravan.test, type = "response")
```

```
Warning message in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
“prediction from a rank-deficient fit may be misleading”
```

```
pred.test2 <- ifelse(probs.test > 0.2, 1, 0)
table(Caravan.test$Purchase, pred.test2)
```

```
   pred.test2
       0    1
  0 4498   35
  1  279   10
```