ISYE 6501 - Homework 1

Question 2.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a classification model would be appropriate. List some (up to 5) predictors that you might use.

Example 1: Transportation Choice When Moving

A classification model could help decide whether to fly or drive when moving from Texas to California. The model would predict which transportation mode someone would choose.

Predictors:

- total travel cost
- total travel time
- driving experience
- amount of luggage
- schedule flexibility

Example 2: Sleep Quality Prediction

Another example is predicting whether I'll have good or bad sleep on a given night based on my daily habits.

Predictors:

- how much I exercised that day
- when I ate my last meal
- how much screen time before bed
- my stress level
- whether I did something relaxing like a massage

Question 2.2.1

Find a good classifier for this dataset using the support vector machine model. Show the equation for your classifier and how well it classifies the data points.

Methodology:

1. I loaded the data and checked that it has 654 rows with 10 predictor variables and 1 response variable (R1).
2. I used the ksvm function from the kernlab package to build the classifier.
3. I tried different C values (0.0001, 0.01, 1, 100, 10000, 1000000) to see which works best. I used scaled=TRUE to normalize the data.
4. To get the coefficients, I used `a = colSums(model@xmatrix[[1]] * model@coef[[1]])` and `a0 = -model@b` (with the negative sign like the question said).

Results:

The model got 86.39% accuracy with C = 0.01, 1, and 100. Here are the results for different C values:

C Value	Accuracy
0.0001	54.74%
0.01	86.39%
1	86.39%
100	86.39%
10,000	86.24%
1,000,000	62.54%

I picked C = 100 since it gave good accuracy. The plot below shows accuracy stays pretty stable from C = 0.01 to 10000. Very small or very large C values don't work well.

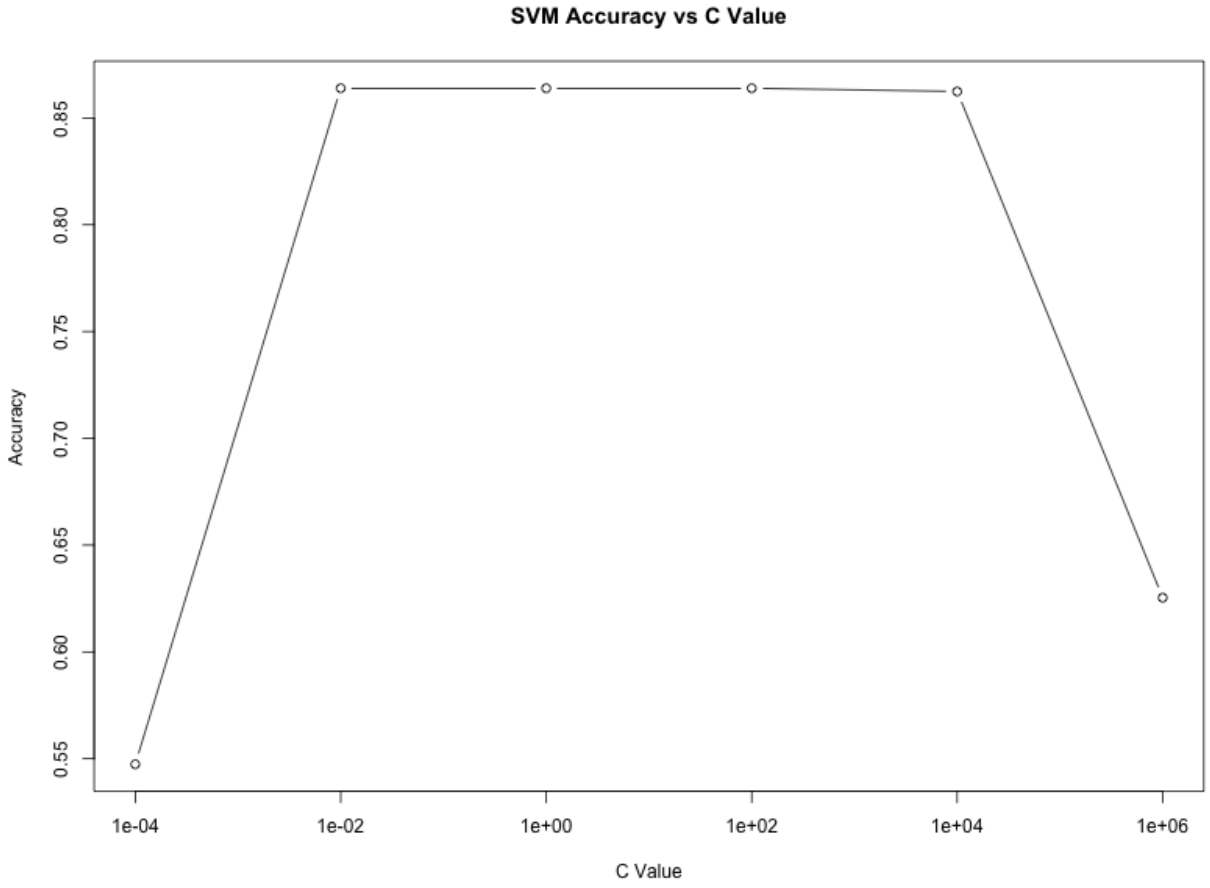

Figure 1: SVM Accuracy vs C Value

The confusion matrix below shows how the model performed on the training data:

The equation for the classifier with C = 100:

```
-0.001007*A1 - 0.001173*A2 - 0.001626*A3 + 0.003006*A8 + 1.004941*A9
- 0.002826*A10 + 0.000260*A11 - 0.000535*A12 - 0.001228*A14 + 0.106364*A15 + 0.081585 = 0
```

Simplified:

$1.005A9 + 0.106\text{A15} + 0.082 = 0$ (approximately)

Looking at the coefficients, A9 (weight ~1.005) and A15 (weight ~0.106) are the most important variables. The other features have very small weights (less than 0.004).

Discussion:

The model gets 86.39% accuracy which seems pretty good. A9 and A15 are clearly the main factors for credit approval. The accuracy is stable for C values between 0.01 and 100. When C is too small (0.0001) or too large (1000000), the accuracy drops a lot.

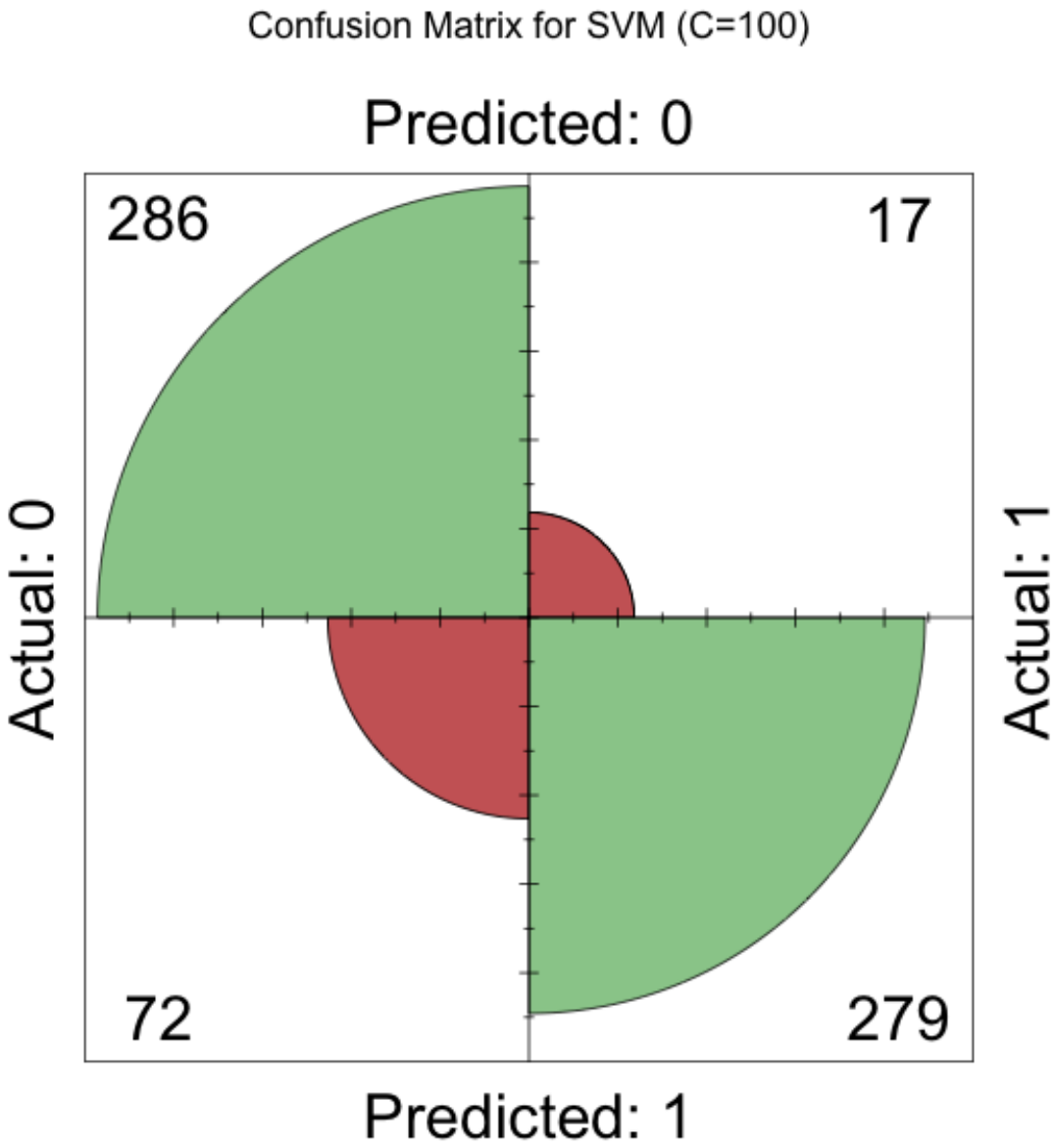

Figure 2: Confusion Matrix for SVM

Question 2.2.3

Using the k-nearest-neighbors classification function kknn, suggest a good value of k and show how well it classifies the data points in the full data set.

Methodology:

1. I converted the data to a data frame with column names A1-A10 for predictors and R1 for the response.
2. I did leave-one-out cross-validation like the question asked - for each point i, I trained on all the data except that point (data[-i,]) and predicted point i.
3. I tested k values of 1, 3, 5, 7, 9, 11, 13, 15, 20, and 25.
4. I used scale=TRUE and converted predictions to 0 or 1 using a 0.5 threshold.

Results:

The best k value was 15 with 85.32% accuracy.

k Value	Accuracy
1	81.50%
3	81.50%
5	85.17%
7	84.71%
9	84.71%
11	85.17%
13	85.17%
15	85.32%
20	85.02%
25	84.56%

So I'd recommend $\mathbf{k = 15}$. The plot below shows accuracy goes up from k=1 to k=15, then drops a bit for larger k values.

Discussion:

The k-NN model with k=15 gets 85.32% accuracy, which is a bit lower than SVM (86.39%). Small k values like 1 and 3 only get 81.50% accuracy because they're too sensitive to noise. As k increases to 15, accuracy improves. After k=15, accuracy drops slightly.

Comparing to SVM:

- SVM got 86.39% accuracy
- k-NN got 85.32% accuracy

SVM performed a little better and it's easier to interpret the coefficients, so I think SVM is the better choice for this dataset.

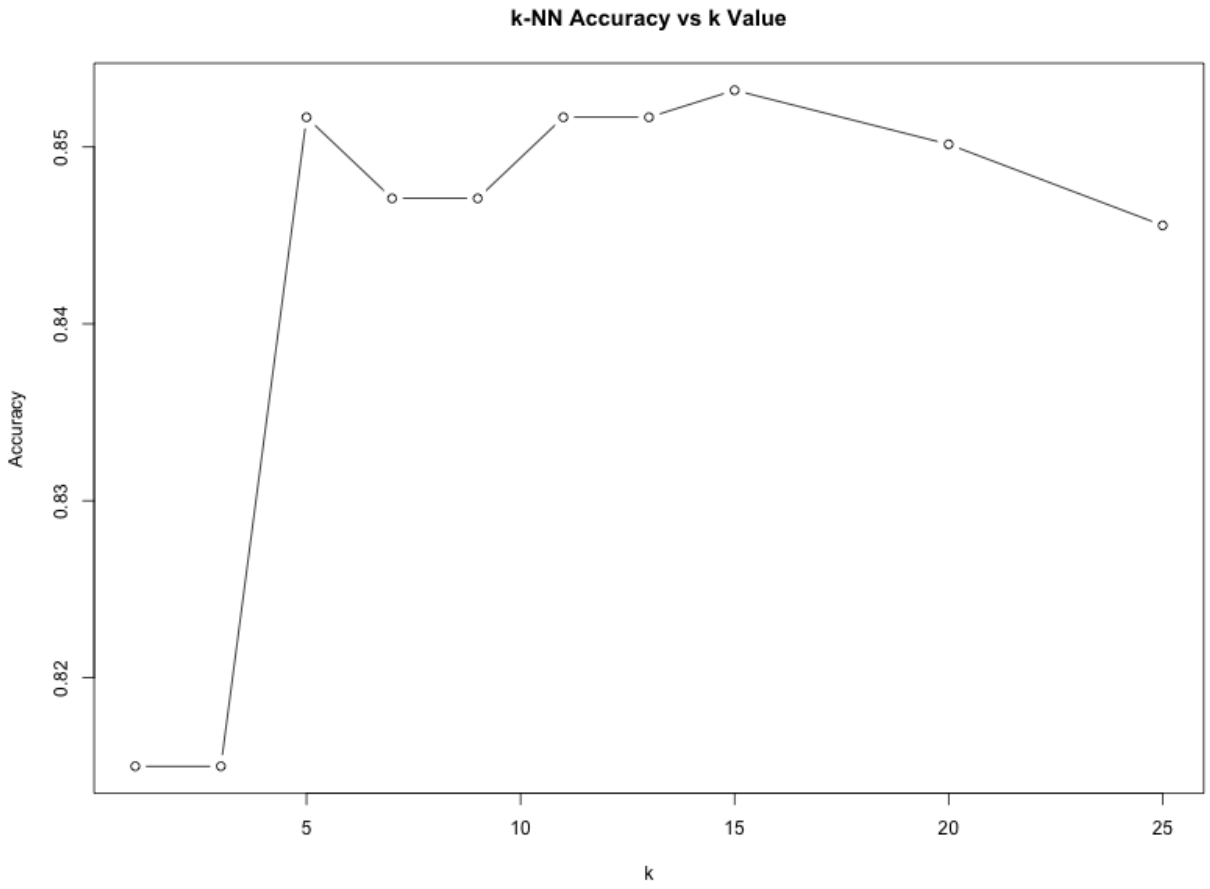

Figure 3: k-NN Accuracy vs k Value