Machine Learning for Public Policy
Mid-term Assignment
JORGE ADRIAN SANCHEZ

Instructions: This is open-book, open-notes, open-internet but no need for any programming to do any of the work. You should show your work instead of just giving me the answer. You can spend as much time as you want. Please submit the assignment on chalk as a pdf.

Section A (Short answers) [35 pts]

1. K-Means clustering algorithm will always find the optimal set of clusters.

   A. True
   B. False

2. Random Forests typically do not require much parameter tuning.

   A. True
   B. False

3. You're asked to predict the probability that the unemployment rate will go down next quarter for each of the neighborhoods in Chicago. Which model would you prefer to use?

   A. Logistic Regression
   B. Support Vector Machines

4. Do you have to do anything special for this problem with the model you chose in #3?
   1. **Scale (normalize) data to not be affected by outliers.**
   2. **Depending on the features, we may transform the variables to be more linear structure.**
   3. **Regularization techniques if we are working with big feature size.**

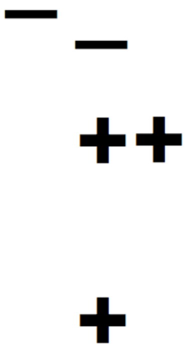
5. What is the training error for a 1-NN classifier?
   0, because we are using assigning a cluster to each point.

6. What is the Leave-one-out cross validation error for k nearest neighbor on the following data set? List any assumptions you may be making.

Assumptions:

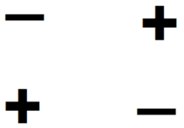
- Euclidean distance
- Majority rule for each cluster
- The two negatives are close together than the two positves and viceversa

A. For k=1 **-> Zero error**
B. For k=3 **-> 40% When using the negative ones you will classified them as wrong. Therefore you will have an error of 2/5.**



7. Which of the following classifiers are appropriate to use for the following data set? Why?

Logistic Regression look to classify linearly. Although you could have data transformation to adapt to the non-linear behavior, this model will perform poorly. In the other hand, with the "kernel trick" you can have non-linear separators with SVM. *Decision Trees* could have the best performance due their capacity of partitioning the instance space into axis parallel regions labeled with a class type.



A. Logistic Regression
B. Decision Trees
C. SVMs

8. In evaluating and generating Association Rules, one of the metrics used is Confidence (defined as the probability of occurrence of B in the basket if the basket already contains A). What is the drawback of only using Confidence as the evaluation metric?

For example let's have the following association rule:

{Milk, Diaper} -> {Beer}
Here X is {Milk, Diaper] -> Y which is {Beer}

Confidence (c) = Measures how often each item in Y appears in transactions that contain X.

In a hypothetical case, confidence can be very high let's say 98%, i.e., when Milk and Diaper are bought together 98% of the times also buy Beer. But what if the support (the percentage of buying milk, diaper and beer) is very low? This means that there are just few purchases as a percentage of the total transactions. You won't put them together because it can be a very very tiny fraction of the transactions.

For this reason, other measures have been proposed as the lift value, the ratio of the observed support to that expected if X and Y were independent.

9. You are being asked to build a model to predict which children in Chicago are at risk of Asthma attacks. You split your data into training and validation sets and experiment with two models, SVMs and Random Forests. SVMs get 100% accuracy on the training set and 70% on the validation set. Random Forests get 80% accuracy on the training set and 75% on the validation set. Which one would you prefer to use?

   A. Random Forests
   B. SVMs

10. The Asthma data has 1000 features (all continuous) but you find out after exploring the data and talking to public health and medical experts that ~10 of them are useful and relevant for the prediction problem. Which of the classifiers below would you choose? And why?

K-NN performance is very dependent on the sparsity of the data and the type of data (in terms of the scales). A second problem is that could be difficult to define the distance measure that can capture variance, the more distant "nearest neighbors" may find themselves in regions (in the instance space) that are already occupied by other classes; as such, they only mislead the classifier. Also, K-nn is difficult to interpret, we would like to have an actual interpretation of how features affect the class.

a. K-NN
b. Decision Trees

11. Does Boosting give you a linear classifier? Why or why not?

No, Boosting, force classifiers to learn about different parts of the input Space, and weigh the votes of different classifiers
It Minimize an equivalent loss fn of a linear regression, where weights are learned incrementally dynamically to fit data, but is not a linear classifier, it can classify non-linear input structures. It is a linear combination of the votes of the different classifiers weighted by their strength.

12. A 311 call center gets incoming calls and needs to route them to one of three departments: Health, Streets & Sanitation, Police. What machine learning method will you use, how, and why?

I will use non-parametric classifier like a decision tree, that recursively partitions the observations into subgroups with a more homogeneous categorical response. This learning algorithm might be very useful given non-linearly associated predictors and are very efficient at selecting from large numbers of predictor variables in contrast with other models like, multinomial logistic Regression (MNL) or SVM, that although their robustness are greatly appreaciated, suffer from the "curse of dimensionality" implicitly necessitating feature selection and balanced training set.

One of the advantages is that one can go with decision trees is its capacity of interpretability by the set of rules such any sequence of tests along the path from the root to a leaf represents an *if-then* rule, and this rule explains why the classifier has labeled a given example with this or that class.

13. What if the routing was now changed from 3 to 30 departments?

I will prefer to use Random Forests, an ensemble model that departs from the previous model (Decision Trees).

Random Forests, combined into a bagged predictor, by letting the N decision trees vote for the most popular class. It shares its capacity to cope with huge features spaces and feature selection is implicitly incorporated during each tree construction, i.e., just those features needed for the test pattern under consideration are involved.

14. You are reading a paper for a new method that you're considering to evaluate in your own work. The reported accuracy is 89.4% and the precision at the top 10% is 56%. Are those numbers high enough to justify you trying the method in your work (please explain your answer in 1-2 sentences)?
    a. Yes
    b. No
    c. Maybe

It depends. If the problem that they are trying to solve is similar to mine, and they are trying to balance the same Error type (I and II), I will use it if accuracy and precision if they are better than mine. It always depend of the type of problem you have and the confidence you want to be.

Section B [50 pts]

1. Decision Trees

Temperature	HomeInsulation	HomeSize	EnergyConsumption
Hot	Poor	Small	Low
Mild	Poor	Medium	High
Cool	Excellent	Large	Low
Hot	Excellent	Large	High
Hot	Excellent	Medium	Low
Mild	Poor	Small	High
Cool	Poor	Small	High
Cool	Excellent	Medium	Low
Cool	Excellent	Medium	High
Cool	Poor	Medium	High

A. What will be the random baseline accuracy for this data set?

Based on this distribution of .6 as class 1 and .4 as class 0 in High Energy consumption.

```
= P(class is 0) * P(you guess 0) + P(class is 1) * P(you guess 1)
= (.6)*(.6) + (.4)*(.4) = .52
```

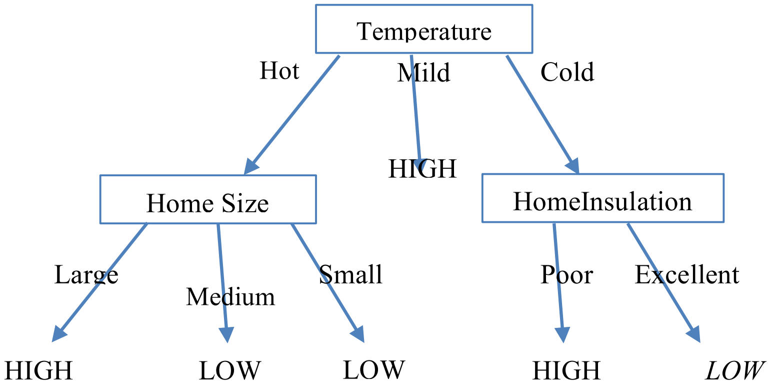
B. Calculate the entropy for the target variable, EnergyConsumption

```
H(T) = -(Ppos)log(Ppos) –(Pneg)log(pneg)
= -(.6)*(np.log2(.6)) - (.4)*(np.log2(.4)) = .9709
```

C. Now calculate the Information Gain if you do a split on the feature "Home Insulation".

```
H(HomeInsulation= Poor) = -(4/5)*(np.log2(4/5)) - (1/5)*(np.log2(1/5)) = .72
H(HomeInsulation= Excellent) = -(2/5)*(np.log2(1/5)) - (3/5)*(np.log2(3/5)) =
.97
H(T, HomeInsulation) = (5/10)*.72 + (5/10)*.97 = .845
I(T, HomeInsulation) = H(T) – H(T, HomeInsulation)= .9709-.845 = .1259
```

D. Using the data above, construct a two-level decision tree that can be used to predict Energy Consumption. Don't worry about overfitting or pruning. You can use a simple algorithm such as ID3 (using information gain as the splitting criterion).



2. Evaluation

The table below shows the predictions of two classifiers, SVM and Logistic Regression for 10 examples. The classifiers are predicting the probability that the Label is 1.

ID	Probability (assigned by SVM)	Probability (assigned by Logistic Regression)	True Label
1	0.98	0.85	1
2	0.2	0.3	0
3	0.1	0.22	0
4	0.99	0.9	1
5	0.55	0.4	0
6	0.05	0.2	0
7	0.4	0.1	1
8	0.35	0.35	0
9	0.65	0.81	0
10	0.75	0.5	1

A. What is the accuracy of the SVM on this set? You will need to make some assumptions here. Be very explicit about your assumptions

We assume that the assignation of the successful class (Label = 1) will be 1 if the probability is >=.5

SVM Confusion matrix

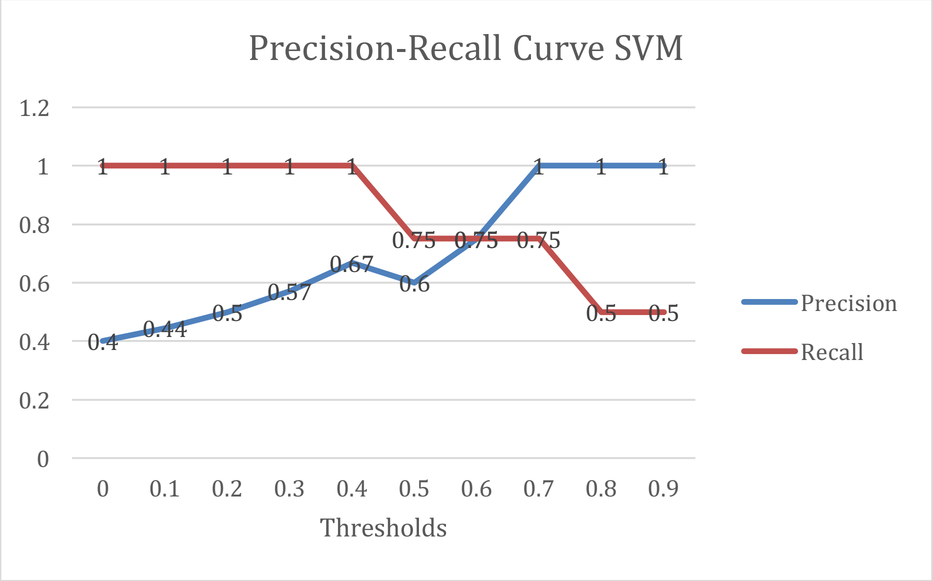
	Pred 0	**Pred 1**
True 0	**4**	**2**
True 1	**1**	**3**

	Pred 0	**Pred 1**
True 0	**TN**	**FP**
True 1	**FN**	**TP**

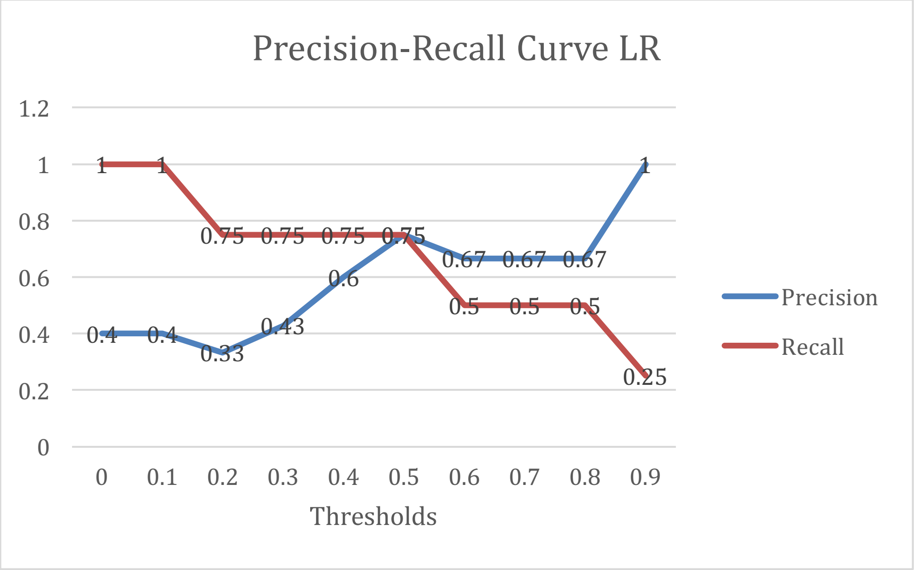
```
Accuracy(SVM) = (TP + TN) / (TP + TN + FP + FN) = 7/ 10
Accuracy(LR) = (TP + TN) / (TP + TN + FP + FN) = 8/10
```

B. Plot the precision recall curves for both classifiers based on these predictions.

```
Precision (SVM) = TP / (TP + FP) = 3/5
Recall (SVM) = TP / (TP + FN) =  3/4
```



```
Precision (LR) = TP / (TP + FP) = 3/4
Recall (LR) = TP / (TP + FN) =  3/4
```



C. Which classifier is better? (again, list the assumptions you're making)

It dependes on the threshold you choose. In this case I choose a threshold of P= >=.5 threshold, in which the Linear Regression has a better, Precision, Recall and Accuracy metrics.

3. You have recently built a model that assigns probability of dropping out to students in grade 9 in a high school. You receive a call from the school administrator asking that “According to your model, Jenny has a score of 50 (out of 100), but I know she is struggling and hasn’t been doing very well in school so far. Why has your model assigned her a score of only 50 and not much higher?”

How would you explain this to the school administrator? Assume this administrator is a smart, educated person with extensive school experience and little or no background in statistics.

Feature composition explanation

First, I will remind him about the processing-making of the model, with special emphasis on the feature selection for the model. It might be that other features are better predictors for dropping out and his just focusing on one.

Outlier explanation

It might be also that Jenny is the outlier of the data. I could also explain that it might be that she has particular characteristics that the model cannot capture and that is the reason why she has a very low score.

Ranking criteria

It might be also true, that there are other students that are in a more worse situation than Jenny. He might know Jenny well, but maybe other students are in a more difficult situation. I will remember that the score criteria is based on a ranking probabilistic model, thus, given a ranking score for each student. It might be important to compare Jenny with other students that have a bigger ranking score.

Data collection

It might be important to check how the information was captured, maybe there was a problem with Jenny or other students regarding the features that might be affected for “struggling and not been doing very well in school” like grades.

Maybe our model is wrong...

In this sense, a domain expert, inspecting the ranking, may decide whether they are intuitively appealing, and whether they agree with his or her “human understanding” of the problem at hand. And maybe that our model is wrong. For instance, the good expertise of the school administrator has pointed out a spurious test that have found their way into the data structure only on account of some random regularity in the data. So, for this reason, I will tell the administrator that our team will check the model performance, given special attention to this particular student.

Then suggest a different way that the administrator can confirm the accuracy of the predictive model you created.

I will suggest a more intuitive way in which the school administrator can check out the performance of the model. I will invite him to randomly select students to be evaluated by the model and compare the result with what the original outcomes. I will show him also, an example of a decision tree, I will go with him through the features split and see how the outcomes would be.

I will thank the time of the school administrator to look detailed at the results at the model. I will confirm that his domain expertise is as important as the work of the data scientist working to resolve this problem with ML.

<u>Section C: Definitions</u> [15 pts]

Give 1-2 sentence definitions of the following terms:

Based on the Introduction to Statistical Learning

1. Regularization

<u>Regularization understood as shrinking method is an alternative for featuring selection. We would like to fit a model containing all p predictors using a technique that constrains or regularizes the coefficient estimates, or equivalently, that shrinks the coefficient estimates towards zero. It may not be immediately obvious why such a constraint should improve the fit, but it turns out that shrinking the coefficient estimates can significantly reduce their variance. The two best-known techniques for shrinking the regression coefficients towards zero are ridge regression and the lasso.</u>

2. Area Under Curve

<u>The ROC curve is a popular graphic for simultaneously displaying the two types of errors for all possible thresholds (TPR and FPR). It is an acronym for receiver operating characteristics. The overall performance of a classifier, summarized over all possible thresholds, is given by the area under the (ROC) curve (AUC). An ideal ROC curve will hug the top left corner, so the larger the AUC the better the classifier. We expect a classifier that performs no better than chance to have an AUC of 0.5 (when evaluated on an independent test set not used in model training). ROC curves are useful for comparing different classifiers, since they take into account all possible thresholds.</u>

3. Over-fitting

<u>Fitting a more flexible model requires estimating a greater number of parameters. These more complex models can lead to a phenomenon known as overfitting the data, which essentially means they follow the errors, or noise, too closely. When a given method yields a small training error but a large test erro, we are said to be overfitting the data.</u>

4. Bootstrap sample

<u>The bootstrap is a widely applicable and extremely powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method. The bootstrap approach allows us to use a computer to emulate the process of obtaining new sample sets. The sampling is performed with replacement, which means that the same observation can occur more than once in the bootstrap data set. We can compute the standard error of these bootstrap estimates.</u>

5. Learning curve

We could think as a performance metric for prediction like *accuracy/error* vs. the *training set size. We should generally see performance improve as the number of training points increases until it r***eaches a plateau**

Section D: Extra Credit [10 pts]

There is a classification method, Perceptron, that we did not cover in class. The assignment here is to read about it, understand it, and tell me how it differs from some of the methods we covered in class, and when would you use it. Would you use it in your project? Why or why not?

We are looking to approximate a linear (hyperplane) boundary for the set of points. The perceptron can be used a classification technique related with logistic regression and can be exemplify as the simple neural network implementation for adjusting weights with a logistic/sigmoid function.
The perceptron is a model that looks to estimate the parameters that define linear classification by gradient descent.

For a model y = mx +B, the idea is to approximate this decision boundary optimizing the values m and B. The gradient descent its base on the idea to find the optimum point in the direction of the gradient (where varies most), minimizes the error of the model and adjusting a with a learning parameter alpha. En each input of data the y= mx+B changes and updates itself to minimize the error for the prediction of the next instance in the training set.

If the solution exists (if its linear separable), it will be found. But whatever the initial weights, the number of attributes, and the learning rate, the perceptron learning algorithm is guaranteed to find a class-separating hyperplane in a finite number of steps—provided, let us not forget, that such class-separating hyperplane exists.

Perceptron is computer-expensive, if the weight-updates are be very small, it will result in slow convergence. *Perceptron learning* is unable to recognize (and eliminate) *redundant attributes*. The learning process will converge to the same weight for both, making them look equally important even though it is clear that only one of them is strictly needed.

I would not use it in my project because of the drawbacks explained before. I would prefer to use a model that can have more insights about which features are important. Second, for the type of data it could be very costly to compute and might never end to find the optimum value because there it might have non-linear characteristics.

You can download a personal python script (html) for a perceptron model at:https://github.com/schzcas/machine-learning-public-policy/blob/master/midterm/gradient-descent_smalldata.html