# CSE Ph.D. Qualifying Exam, Fall 2021

## Data Analysis

Please answer three of the following four questions. All questions are graded on a scale of 10. If you answer all four, all answers will be graded and the three lowest scores will be used in computing your total. Show all your work and write in a readable way.

1. **SVMs and the Kernel trick**

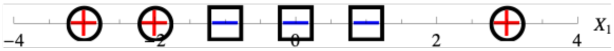


Figure 1: Dataset for SVMs and Kernels.

You are given a data set $D$ (see Fig. 1) with data from a single feature $X_1$ in $\mathbb{R}^1$ and corresponding label $Y \in \{+, -\}$. The data set contains three positive examples at $X_1 = \{-3, -2, 3\}$ and three negative examples at $X_1 = \{-1, 0, 1\}$.

(a) (1 point) Can this data set (in its current feature space) be perfectly separated using a linear separator? Why or why not? (Explain in 1 line)

(b) (2 points) Lets define the simple feature map $\phi(u) = (u, u^2)$ which transforms points in $\mathbb{R}^1$ to points in $\mathbb{R}^2$. Apply $\phi$ to the data and plot the points in the new $\mathbb{R}^2$ feature space (i.e. just show the plot). Can a linear separator perfectly separate the points in the new $\mathbb{R}^2$ features space induced by $\phi$? Why or why not? (Again, explain in 1 line)

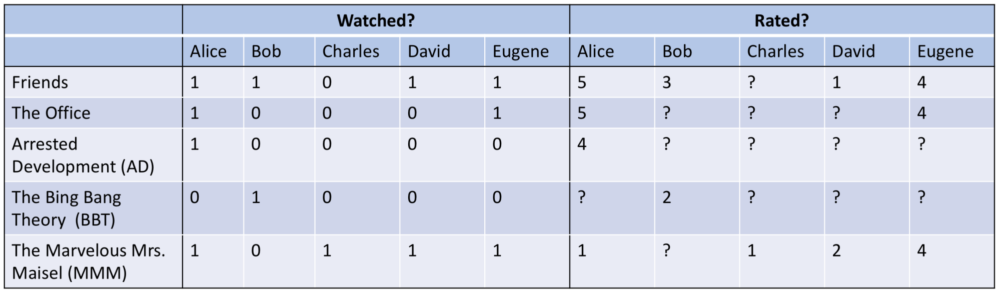
(c) (1 point) Give the analytic form of the kernel that corresponds to the feature map $\phi$ in terms of only $X_1$ and $X_2$. Specifically define $k(X_1, X_2) =< \phi(X_1), \phi(X_2) >$ ($< ., . >$ is the dot-product of two vectors), and give the analytical form of $k(., .)$.

(d) (4 points) Construct a maximum-margin separating hyperplane. This hyperplane will be a line in $\mathbb{R}^2$, which can be parameterized by its normal equation, i.e. $w_1Y_1 + w_2Y_2 + c = 0$ for appropriate choices of $w_1$, $w_2$ and $c$. Here, $(Y_1, Y_2) = \phi(X_1)$ is the result of applying the feature map $\phi$ to the original feature $X_1$. Give the values for $w_1$, $w_2$ and $c$. Also, explicitly compute the margin for your hyperplane. You do not need to solve a quadratic program to find the maximum margin hyperplane. Instead, let your geometric intuition guide you.

(e) (2 points) Draw the decision boundary separating of the separating hyperplane, in the original $\mathbb{R}^1$ feature space. Also circle the support vectors.

2. **Recommendation Systems**

You have collected the following ratings of popular comedy TV shows from five users:



| | Watched? | | | | | Rated? | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Alice | Bob | Charles | David | Eugene | Alice | Bob | Charles | David | Eugene |
| Friends | 1 | 1 | 0 | 1 | 1 | 5 | 3 | ? | 1 | 4 |
| The Office | 1 | 0 | 0 | 0 | 1 | 5 | ? | ? | ? | 4 |
| Arrested Development (AD) | 1 | 0 | 0 | 0 | 0 | 4 | ? | ? | ? | ? |
| The Bing Bang Theory (BBT) | 0 | 1 | 0 | 0 | 0 | ? | 2 | ? | ? | ? |
| The Marvelous Mrs. Maisel (MMM) | 1 | 0 | 1 | 1 | 1 | 1 | ? | 1 | 2 | 4 |

Figure 2: TV Shows Rating Matrix.

(a) (3 points) To generate recommendations, you adopt the following policy: "if a user U likes item X, then U will also like item Y". You implement this by *maximizing the cosine similarity* between the ratings of items X and Y. Your policy also states that you will only make a recommendation to user U if (a) U has not already watched or rated Y and (b) U's rating of item X is at least 3.

Using this policy, which TV show would be recommended to Eugene? Show the comparisons that you made.

(b) (3 points) Next, you design a recommendation system to rank TV show to find the 'Best TV Shows of All Times', using the following formula: *ratings(i)* = *a* + *b(i)*. In this formula, you set *a* as a global term and *b(i)* as an item's bias score. You first fit this model to calculate *a* as the mean of all ratings across the dataset, and in the process, you calculate *b(i)* to be the remainder value per item.

You rank the items according to their bias scores (higher bias score is ranked higher). Which item, among the five shows shown in Table 1, would be the Best TV Show and which one would be the Worst TV show? Show your calculations.

(c) (2 points) An attacker aims to manipulate the above recommendation system trained on the TV Show Rating data. The recommendation algorithm generates recommendations based on bias score of the item (higher rated items are ranked higher). The goal of the attacker is to increase the ranking of the lowest ranked item as you identified in the previous question.

To conduct the attack, the attacker can adopt one of the two strategies: first, the attacker can change the existing ratings in the data, and second, the attacker can create new account(s) and give ratings.

Of the two strategies, which one is:
(i) more likely to succeed?
(ii) more easily detectable by an attacker detection system?
(iii) has lower cost?

(d) (2 points) You come up with the idea of training a deep learning-based recommendation system model, namely the Neural Collaborative Filtering (NCF) model, on your large dataset to create better recommendation models. Your large dataset has 10 million ratings given by approximately 100,000 users to approximately 1,000,000 movies.

Your NCF model first generates 8-dimensional user and item embeddings. Then you pass the embeddings through two fully-connected neural CF layers with sizes 8x16 and 16x16 dimensions. Finally, this is passed through a 16x1 output layer with ReLU activation to produce a single prediction value of recommending an item to a user. You train the model for 10 epochs with back-propagation.

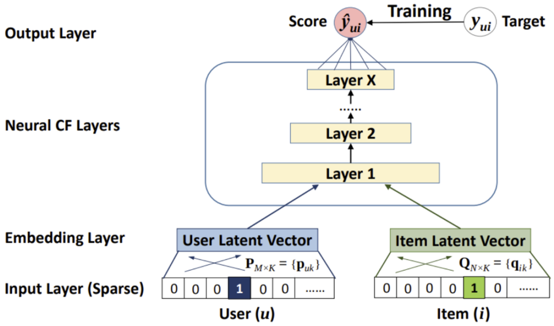


Figure 3: The Neural Collaborative Filtering (NCF) model.

After training the model, you find that the model does not perform well. What changes can you make to the model or parameters to potentially improve the performance? Give at least three options. Note that you cannot choose a different model now.

3. **Classification: ROC, random forests, interpretation**

Holly and Will are cyber analysts working at a large technology company. They are developing binary classification techniques to detect network security threats.

(a) (1 point) In the analysts' context, what would the *positive* class refer to?

(b) (1 point) The analysts are consider using an ROC (receiver operating characteristic) curve to visualize the classifier's performance. What are the two axes in a ROC plot?

(c) (2 points) How is the ROC curve of a classifier generated? In other words, how should the analyst generates the points from which they can link together to build the curve? For easier discussion, your answer may center around a binary classifier of your choice. You are welcome to include illustrations to support your answer.

(d) (1 point) If a classifier performs perfectly (i.e., it makes no mistakes), where will its "point" be located on the ROC plot?

(e) (1 points) On the same plot, the analysts want to draw both the ROC curve of their classifier, and that of a "baseline" classifier that guesses the positive class half of the time, which is a straight line that goes diagonally from the plot's bottom left corner to the upper right corner. Is it possible for the ROC curve of the analysts' classifier to lie *completely under* the baseline curve? If yes, when would that happen? If no, why not?

As Holly and Will are evaluating more classification approaches, they come across random forests.

(f) (2 points) Random forests is a modification over bagging decision trees. The random forests improves variance reduction (over bagging) by reducing correlation among trees. Briefly explain how this correlation reduction ("de-correlation") among trees is achieved when growing the trees.

(g) (2 points) The analysts debate whether a random forest is an "interpretable" model. Holly argues that it is interpretable, while Will argues that its interpretability is limited. Briefly discuss why they may both be correct.

4. **Text Representation Learning and Gradient Descent**

We consider the problem of learning word vectors from an unlabeled corpus using the skipgram model. Word embedding techniques learn to represent the words in a large text corpus as $N$ dimensional vectors, with the goal of making similar words close to each other in the vector space. The well-known skip-gram model achieves this goal by predicting the context words for a given center word. Let $\boldsymbol{v}_c$ denote the word vector of a given center word $\boldsymbol{c}$. Skip-gram models the probability of observing a context word $\boldsymbol{o}$ from the vocabulary using the softmax function:

$$p(\boldsymbol{o} \mid \boldsymbol{c}) = \frac{\exp\left(\boldsymbol{u}_o^\top \boldsymbol{v}_c\right)}{\sum_{w=1}^{W} \exp\left(\boldsymbol{u}_w^\top \boldsymbol{v}_c\right)}, \tag{1}$$

where $\boldsymbol{w}$ is the w-th word in the vocabulary and $\boldsymbol{u}_w (w = 1, \ldots, W)$ are the context word vectors.

However, instead of directly maximizing the likelihood of groundtruth context words, we often use the *negative sampling* technique for learning the word vectors. It randomly draws $K$ negative samples (words) from the vocabulary, denoted as $1, \cdots, K$ $(o \notin \{1, \ldots, K\})$. The learning objective for negative sampling is to distinguish the groundtruth context word $\boldsymbol{o}$ from the negative samples $1, \cdots, K$. The loss function for this negative sampling model is given by:

$$J\left(\boldsymbol{o}, \boldsymbol{v}_c, \boldsymbol{U}\right) = -\log\left(\sigma\left(\boldsymbol{u}_o^\top \boldsymbol{v}_c\right)\right) - \sum_{k=1}^{K} \log\left(\sigma\left(-\boldsymbol{u}_k^\top \boldsymbol{v}_c\right)\right), \tag{2}$$

where $\sigma(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function, and $\boldsymbol{U}$ is the set of embeddings for all the words in the vocabulary.

(a) (5 points) Derive the stochastic gradient descent algorithm for the unknown parameters $\boldsymbol{v}_c$, $\boldsymbol{u}_o$ and $\boldsymbol{u}_k$ $(k = 1, 2, \ldots, K)$ for the loss function $J$.

(b) (3 points) From your derived gradient descent algorithm, discuss why using Eq. (2) and the gradient descent procedure can make semantically similar worlds close to each other in the vector space.

(c) (2 points) Discuss the advantage of using Eq. (2) instead of Eq. (1) for learning word embeddings.