**Final Exam Quiz 1 of 3 (Classification)**

* **Due** Sep 24 at 11:59pm

* **Points** 34

* **Questions** 10

* **Available** Sep 22 at 8am - Sep 24 at 11:59pm

* **Time Limit** 30 Minutes

**Instructions**

You may use any resource you would like to help you complete this quiz (notes, internet, textbook), aside from another person. This quiz will cover the classification concepts covered in Project 5 and the following three lectures:

* [14-classification-kNN.ipynb](https://elearning.mines.edu/courses/52392/files/5459044/download?wrap=1)[Download 14-classification-kNN.ipynb](https://elearning.mines.edu/courses/52392/files/5459044/download?download_frd=1)
* [16-classification-svm.ipynb](https://elearning.mines.edu/courses/52392/files/5459171/download?wrap=1)[Download 16-classification-svm.ipynb](https://elearning.mines.edu/courses/52392/files/5459171/download?download_frd=1)
* [17-model\_selection.ipynb](https://elearning.mines.edu/courses/52392/files/5459097/download?wrap=1)[Download 17-model\_selection.ipynb](https://elearning.mines.edu/courses/52392/files/5459097/download?download_frd=1)

Good luck.

This quiz was locked Sep 24 at 11:59pm.

Attempt History

|  | **Attempt** | **Time** | **Score** |
| --- | --- | --- | --- |
| **LATEST** | [Attempt 1](https://elearning.mines.edu/courses/52392/quizzes/74260/history?version=1) | 25 minutes | 24 out of 34 \* |

*\** Some questions not yet graded

Score for this quiz: **24** out of 34 \*

Submitted Sep 22 at 10:29am

This attempt took 25 minutes.

**Question 1**

**2 / 2 pts**

In classification, we typically try to predict                                , assigning points to groups.

**Correct!**

categorical responses

Yes, we predict categorical responses or class labels. Our classification training data has the same kinds of inputs as in regression, but the target variable is discrete, not continuous.

e.g., The training data might include features about a borrower's financial history, credit score, owning/renting history, etc. The target might be an indicator on whether or not the borrower ultimately defaulted on their loan. The learning task is to find a classifier function which will let us predict future defaults given the same information about a potential borrower.

continuous outputs

observations

features

**Question 2**

**2 / 2 pts**

One of the simplest techniques is k-nearest neighbor classification. This classifier uses a \_\_\_\_\_\_\_\_\_\_\_\_ strategy to determine class assignment of a new unseen data observation.

guessing

**Correct!**

voting

When we get a new data point we haven't seen before, we determine what class to assign it to by simply looking at the k nearest points and taking a majority vote on the class (and recall, ties are sometimes broken randomly)!

min/max

**Question 3**

**4 / 4 pts**

What happens as we decrease or increase k in k-nearest neighbor classification?

It does not affect overfitting or underfitting.

You could underfit by using too small of a k and overfit by using too high of a k.

**Correct!**

You could overfit by using too small of a k and underfit by using too high of a k.

Classification is sensitive to the complexity of your model, In k-nearest neighbor classification, you overfit by using too few neighbors, and underfit by using too many neighbors.

**Question 4**

**2 / 2 pts**

We used the Support Vector Machine (SVM) in our Classification project, the SVM is a Maximum margin classifier.

**Correct!**

True

True, we used the SVM for classification, and the model maximizes the distance from the decision boundary for each example.

False

**Question 5**

**0 / 4 pts**

If the number of features is large as compared to the number of observations, is it a good idea to reduce dimensionality before fitting the SVM?

No, we should not

**You Answered**

Yes, we should

**Correct Answer**

Does not matter, this algorithm can work well with high-dimensional data, even data with more dimensions than samples.

**Question 6**

**4 / 4 pts**

The purpose of defining a data-appropriate kernel function is to take data as input and transform it into a more suitable form. What are possible kernel functions? (Check all that apply.)

**Correct!**

RBF or Gaussian

Random

**Correct!**

Linear

**Correct!**

Polynomial

Blank

Uniform

**Question 7**

**4 / 4 pts**

We worked with / tuned two parameters other than the kernel function. Match the statements below with the appropriate parameter.

**Correct!**

**controls trade-off between bias and variance.**

               C

**Correct!**

**controls the shape of the peaks/smoothens out the Gaussian.**

               Gamma

**Correct!**

**plays the part of penalizing misclassifications.**

               C

**Correct!**

**is only used with the RBF kernel.**

               Gamma

Other Incorrect Match Options:

* Alpha
* Beta

**Question 8**

**4 / 4 pts**

The evaluation metrics we have used for classification can be visualized using the Confusion Matrix, match the following terms:  
Fill in the squares with: FP, FN, TP, TN.  
Add to the squares: Miss, Hit, False Alarm, Correct Reject

**Correct!**

**False Positive**

               False Alarm

**Correct!**

**False Negative**

               Miss

**Correct!**

**True Positive**

               Hit

**Correct!**

**True Negative**

               Correct Reject

**Question 9**

**2 / 2 pts**

The only metric that really matters is Accuracy!

True

**Correct!**

False

Although accuracy is a good measure of classifier performance, it could give misleading results at times (e.g., when TP < FP). You could explore some of the other metrics, like precision, recall, f-score, for a more robust measure.

**Question 10**

**Not yet graded / 6 pts**

We also studied k-fold cross validation, please discuss this technique and include the following:

- what does k stand for?  
- when training and testing with one dataset, is a test/train split first still necessary to evaluate accuracy when using k-fold cross validation?  
- what are the advantages of using k-fold cross validation?

Your Answer:

**k:** the number of groups the data is split into. you split the data into k parts (folds)

**test/train split:**is not necessary before a k-fold. The k-fold is essentially performing multiple test/train splits, shifting the train/test data based on your folds.

**Advantages:**

* Better score estimates
  + All data gets in the test set (once)
  + All data gets in the training set (multiple times)
* Some information about sensitivity of the data to particular splits
* Can use more data in training (e.g., in 5-fold, use 80% of data for each training run)

- k stands for the number of folds.

- no, we do not still need to do a train/test split since the goal of cross validation is to make better use of our (usually limited) training data and to reduce bias introduced by the training/test split itself.

- The basic idea is to utilize multiple training/test splits and then compute a composite score from each individual run.

Quiz Score: **24** out of 34