**PROJECT PLAN**

1. Students’ name and Purdue e-mail. The project is to be done in groups of 4 students.

[furlonga@purdue.edu](mailto:furlonga@purdue.edu)

[apolcyn@purdue.edu](mailto:apolcyn@purdue.edu)

[sun594@purdue.edu](mailto:sun594@purdue.edu)

[aarjunak@purdue.edu](mailto:aarjunak@purdue.edu)

1. [Up to 3 lines] Definition of the problem, possibly relevant to your interests.

We will be writing an algorithm which looks at a dataset of breast cancer tumor readings from Wisconsin and predicts whether the tumor in question is malignant or benign. Our hope is to create an algorithm that can accurately differentiate a malignant tumor from a benign tumor in order to make diagnosis of this disease quicker.

1. [Up to 3 lines] Description of the dataset (or datasets) to be used. Datasets should be already publicly available, since there is not enough time for you to collect data. For possible datasets, see the course webpage.

Our dataset contains 569 instances of tumors with 32 features each. Each tumor is given a diagnosis as well as many readings which give the dimensions of the tumor amongst other readings. The dataset is already in table format with all numeric values, making it more trivial for our algorithm to work with it.

1. URL where the above dataset(s) is(are) available.

https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

1. [Up to 5 lines] Which machine learning algorithm(s) is(are) going to be used? (e.g., SVM, Prank, Adaboost, etc.) **You are allowed to either implement this from scratch or use third-party code, e.g., liblinear for SVM.**

We will be using a dual kernel and a SVM with no kernel. We will be testing different degrees of polynomials for our kernel algorithm to determine what works best with our dataset. Our group is currently looking into which third-party code for these types of algorithms best fits our needs and will be making that decision soon after further research.

1. [Up to 5 lines] Which cross-validation technique(s) is(are) going to be used? (e.g., training/validation/testing, k-fold cross-validation, bootstrapping). **You MUST implement this from scratch.**

We will use k-fold cross-validation in order to get a more holistic view of the accuracy of our model. Since certain types and properties of cancer might be rare, we want to be sure that our trained model performs well. Therefore, we want to test the model across as many data points as possible.

1. [Up to 10 lines] Which hyperparameter(s) is(are) going to be tuned. **You MUST implement this from scratch.**

We cannot assume that tumor data will be linearly separable, so we will need to use slack variables. We intend to weight a diagnosis more heavily towards a malignant diagnosis, as a false positive can be looked into further while a false negative could mean a cancer going undiagnosed and leading to irreparable damage. While we are tuning our algorithm, we will be sure to try many different values of our hyperparameters to ensure this is done accurately.

1. [Up to 15 lines] Which experimental results will you show? (e.g., plots of number of samples versus accuracy (you can use different subsets of the same dataset), regularization parameter versus accuracy, ROC curves, plots of different datasets, etc.) **You MUST implement this from scratch.**

Because we are particularly concerned with false negatives we will run an ROC curve analysis in order to properly determine how likely it is that we make an incorrect diagnosis. We will also plot the number of samples versus the accuracy of our algorithm on the test dataset so that we may get a general sense of the accuracy of our diagnoses. We will also include a plot to compare our training and test errors.

1. Which programming language are you going to use? (Only MATLAB, C++, Java and Python are allowed.)

Python 3

**Advice: Do not spend too much time on things such as "understanding the data", "memory problems because your data is too big", etc. Only if you are already familiar with computer vision, brain data, natural language processing, big data, parallelism, etc. then you can make use of those things, but this will not imply that you will get a higher grade just based on that fact. In general, I would recommend using easy-to-understand datasets, and smaller subsets of the data, for instance.**