**Instructions**: Please complete and submit your work to the appropriate folder in LumiNUS. You may work in study groups, but each student must be responsible for their own submission.

Please submit all the following documents as a single zip file named StudentID-Name-HW1.zip:

1. Completed Word file named as StudentID-Name-HW3.docx (with all results)
2. Print preview of ipynb file named as StudentID-Name-HW3.pdf (with results)
3. Working ipynb file named as StudentID-Name-HW3.ipynb
4. Consider building an SVM classifier for the following two-class training data:

Positive class: { (-1, 3) (0, 2) (0, 1) (0, 0) }; Negative class: { (1, 5) (1, 6) (3, 3) }

1. Plot the training points. Use ‘+’ for positive class and ‘o’ for the negative class.
2. By inspection, draw a linear classifier that separates the data with maximum margin.
3. The linear SVM is parameterized by h(x) = (**w**^t)(x) + b. What are the parameters **w** and b for this problem?
4. Suppose you observe an additional set of points, all from the positive class.

Additional data points in positive class: { (−2, 0) (−2, 1) (−2, 3) (−1, 0) (−1, 1) }

A picture containing clock

Description automatically generatedWhat is the linear SVM (in terms of **w** and b) now?

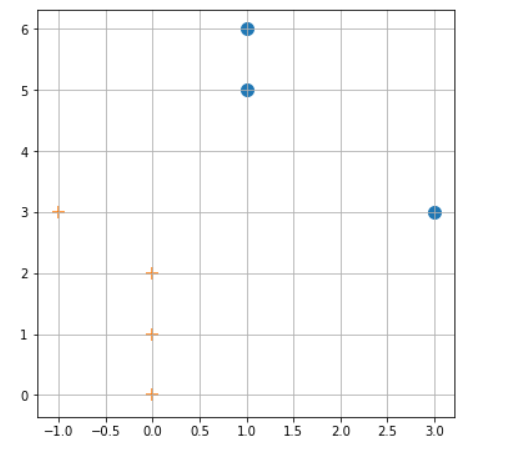
1. Consider the dataset on the right. Consider using the SVM with soft margin classifier with parameter C.
   1. Draw the linear classifier when C is large.
   2. Draw the linear classifier when C is small.
   3. Which value of C yields the classifier most closely resembling the hard margin SVM solution?
   4. Using your two examples, explain how the C parameter helps with overfitting in SVMs.
2. In this problem, we will look at the Breast Cancer Wisconsin (Diagnostic) Data Set available UCI Machine Learning Repository. Please use the wdbc.data dataset from:

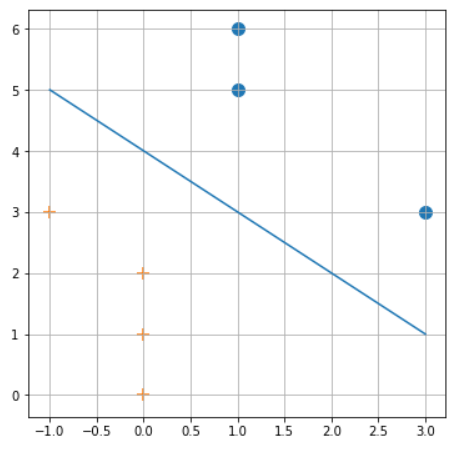
https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

* Compute the performance of the SVM algorithm on this dataset for predicting the whether the cancer is malignant or benign. Use a random train/test data split of 70%/30%. Repeat this process 20 times and compute the average performance.
* Please evaluate the following algorithms:
* SVM1: SVM with linear kernel
* SVM2: SVM with RBF kernel
* SVM3: Same as SVM2 but with regularization (soft margin), vary C and report your best results.
* Please compute the following metrics and fill in the table below.
* Training Accuracy and Test Accuracy
* Precision and Recall (which are important metrics that complement Accuracy)
* You can read about performance metrics at: <https://en.wikipedia.org/wiki/Confusion_matrix>
* SKLearn contains functions to compute these metrics:

<https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Accuracy | | Precision | Recall |
|  | Train | Test |  |  |
| SVM1 |  |  |  |  |
| SVM2 |  |  |  |  |
| SVM3  C = |  |  |  |  |

1. (a) 

(b) 

By inspection, a linear classifier (blue colored line) that separates the data with maximum margin is as drawn in the above image.

y = mx + b is the equation of a straight line

From the graph, the blue colored line (linear classifier) cuts through (-1,5) and (3,1)

5 = m(-1) + b

-m + b = 5 – (1)

1 = m(3) + b

3m + b = 1 – (2)

Solving (1) and (2) simultaneously,

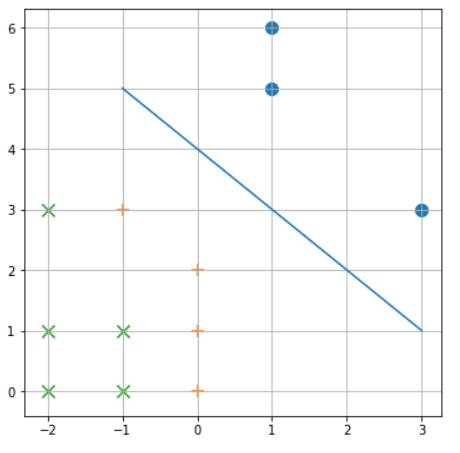
m = -1

b = 4

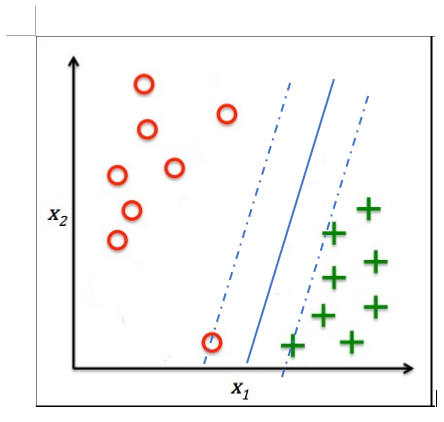
Therefore, y = -x + 4 is the equation of linear classifier based on inspection of the data points provided.

(c) A picture containing text, computer, keyboard, desk

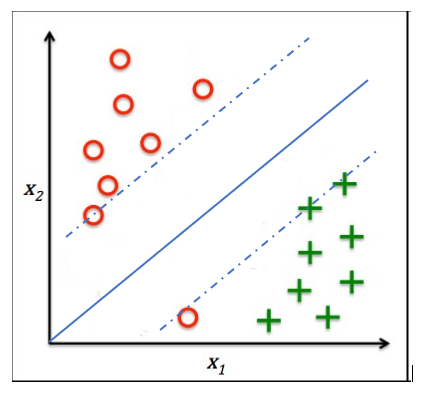
Description automatically generated

(d) 

There is no change in **w** or b. **w** is still (-0.5, -0.5) and b is still 2. This is because after the addition of new data points, the nearest data points to the linear classifier (support vectors) remains the same as before, that is, points (3,3), (0, 2) and (1, 5).

1. (a) 

The solid blue line represents the linear classifier.

(b) 

The solid blue line represents the linear classifier.

(c) A high value of C yields the classifier most closely resembling the hard margin SVM solution.

(d) The C parameter can be viewed as a way to control overfitting in SVM. It trades off the relative importance of maximizing the margin and fitting the training data. In this example, in comparison to a small C parameter, a large C parameter refers to more penalty for misclassifications, which means resulting in less outliers in the plot. This leads to smaller margins but less misclassifications, compared to small C parameter. Thus, in this case, the SVM algorithm with large C parameter imposes large penalty for misclassification and tries to correctly classify an outlier with noise. This is essentially overfitting.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | | **Precision** | **Recall** |
|  | **Train** | **Test** |  |  |
| **SVM1** | 1 | 0.944444 | 0.924518 | 0.933407 |
| **SVM2** | 1 | 0.961696 | 0.956487 | 0.943459 |
| **SVM3**  **C = 10** | 0.99196 | 0.971345 | 0.966997 | 0.958738 |

For SVM3, we vary the value of C and the best result obtained is when C = 10.