1. Classification

$y \in \{1, 2, 3, .., k\}$

Task make a function, that separates known classes.

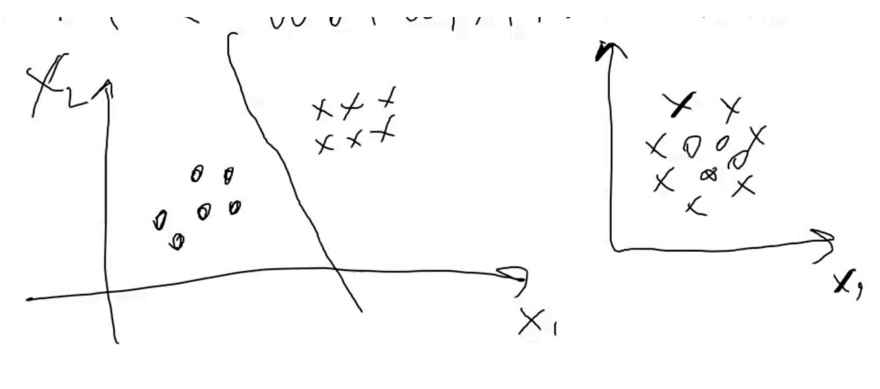

Figure 1: Classification visualization

Accordingly to image, linear regression is not suitable for this type of task(especially right)

Firstly, we will solve binary classification task$\{0, 1\}$. Model will have 1 output - probability of x is from class 1.

1.1. Logistic regression

Logistic regression type of regression that predicts a probability of an outcome given one or more independent variables. With a threshold returned probability can be mapped to a discrete value.

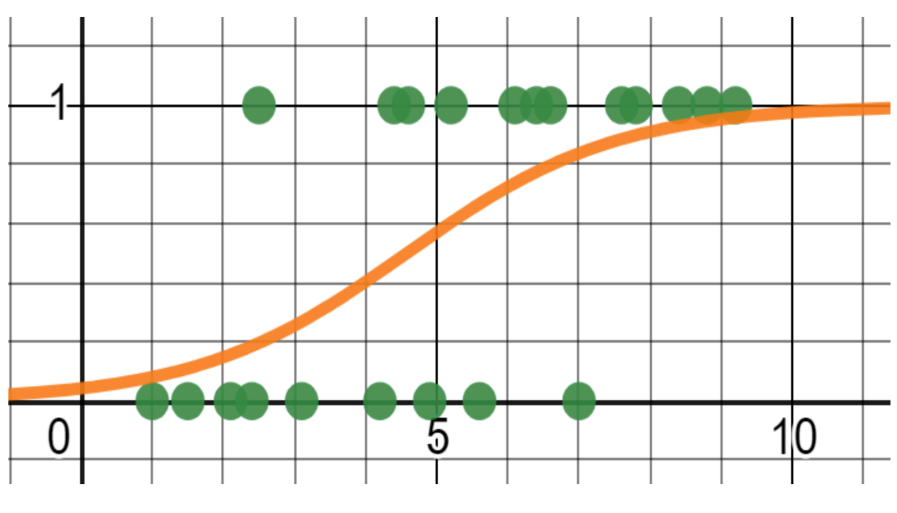

Figure 2: Logistic regression

1.1.1. Formula

Logistic regression is a S-shaped curve:

$$y = \frac{1}{1 + e^{-\sum_{i=1}^{m} \Theta_i X_i + \Theta_0}}$$

1.1.2. Loss function

BCE(Binary cross entropy) loss function

$$\begin{cases} -\log(p_i), y_i = 1 \\ -\log(1 - p_i), y_i = 0 \end{cases}$$

p_i - model output probability for i example. (class 1)

$$\text{BCE} = -y_i \log(p_i) - (1 - y_i) \log(1 - p_i)$$

$$\begin{cases} \text{TP } y_i, p_i = \{1, 1\} \text{ BSE} = 0 \\ \text{TN } y_i, p_i = \{0, 0\} \text{ BSE} = 0 \\ \text{FN } y_i, p_i = \{1, 0\} \text{ BSE} \to \text{inf} \\ \text{FP } y_i, p_i = \{0, 1\} \text{ BSE} \to \text{inf} \end{cases}$$

Loss for gradient:

$$L = -\frac{1}{N} \sum_{i=1}^{n} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \longrightarrow \min$$

$$\frac{\partial f}{\partial \Theta_j} = -\frac{1}{N} \sum_{i=1}^{n} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i))'$$

1.2. Classification Metrics

With MINST dataset the model which gives 90% accuracy of 5 classification = random model.

The solution is to use confusion matrix - is a table that is used to define the performance of a classification algorithm(TP, FP, TN, FN). Each row in a confusion matrix represents an actual class, while each column represents a predicted class.

	Predicted	Predicted
Class A	TP	FP
Class B	FN	TN

null-hypothesis is that all patients are in False group. The rejecting of null-hypothesis is stating that patient is in True group. The failure of rejecting NH - false positive. The failure of accepting HN - false negative.

positive = rejecting null-hypothesis

precision - how much false positive mix you would have

$$\text{precision} = \frac{\text{TP}}{\text{TP + FP}}$$

precision is not enough - the model can make only 1 true prediction -> 100% precision

Як багато визначених тестом релевантних елементів справді релевантні.

recall - how much false negative mix you would have. The coverage of TP.

$$\text{recall} = \frac{\text{TP}}{\text{TP + FN}}$$

sensitivity - True positive rate. Is the probability of a positive test result, conditioned on the individual truly being positive. Same as recall. TP/(TP+FN)

Як багато релевантних елементів було визначено тестом як релевантні. A test with a higher sensitivity has a lower type II error rate(false negative).

specificity - True Negative Rate. TN/(FP+TN). The same, P of detect False of all observations=False.

Як багато негативних елементів було визначено тестом як негативні. A test with a higher specificity has a lower type I error rate(false positive).

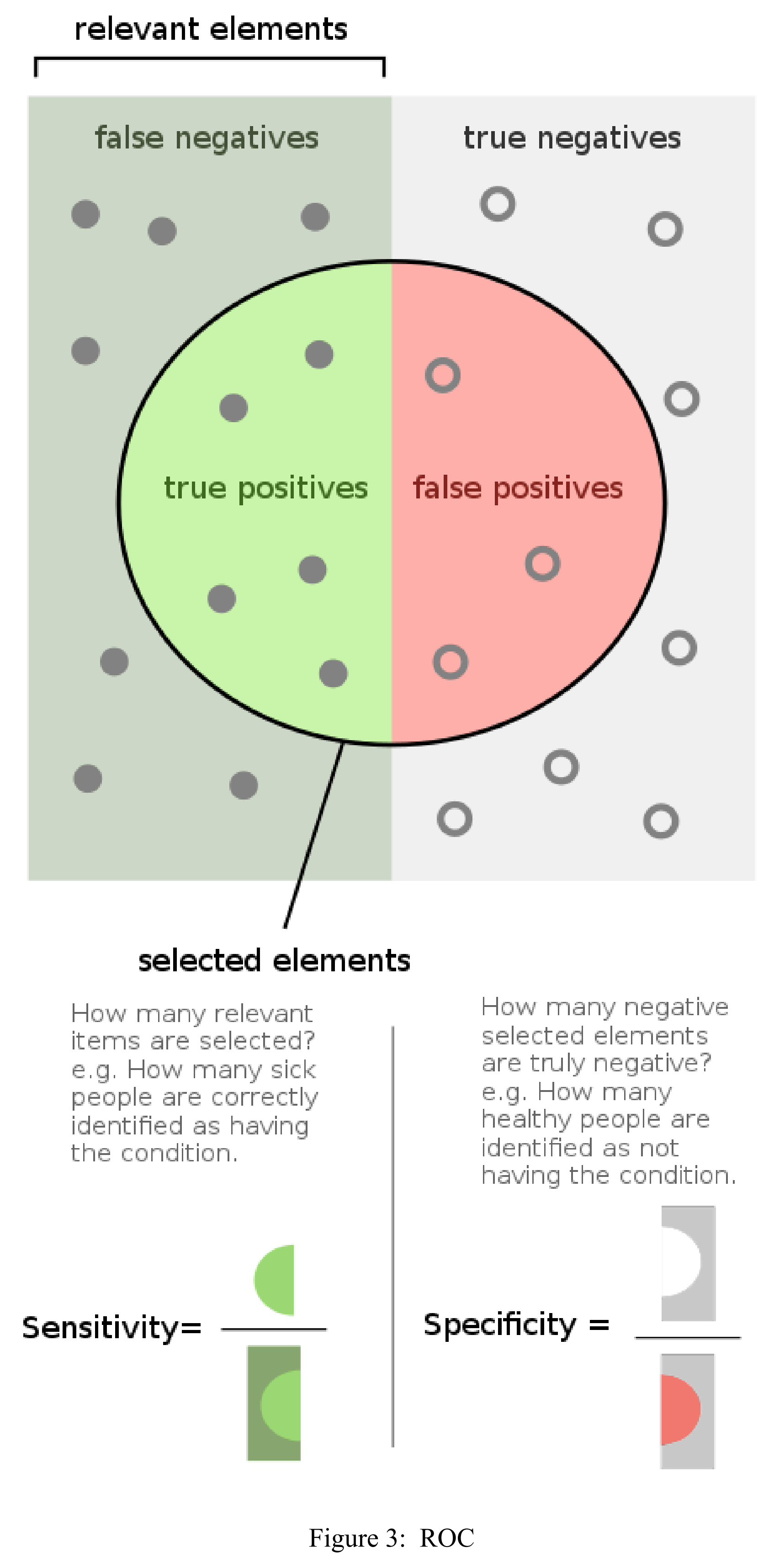

Figure 3: ROC

1.3. ROC

FPR(false alarm ratio) = FP/(FP + TN). Probability of falsely rejecting null-hypothesis. False Positive Rate. = 1 - specificity.

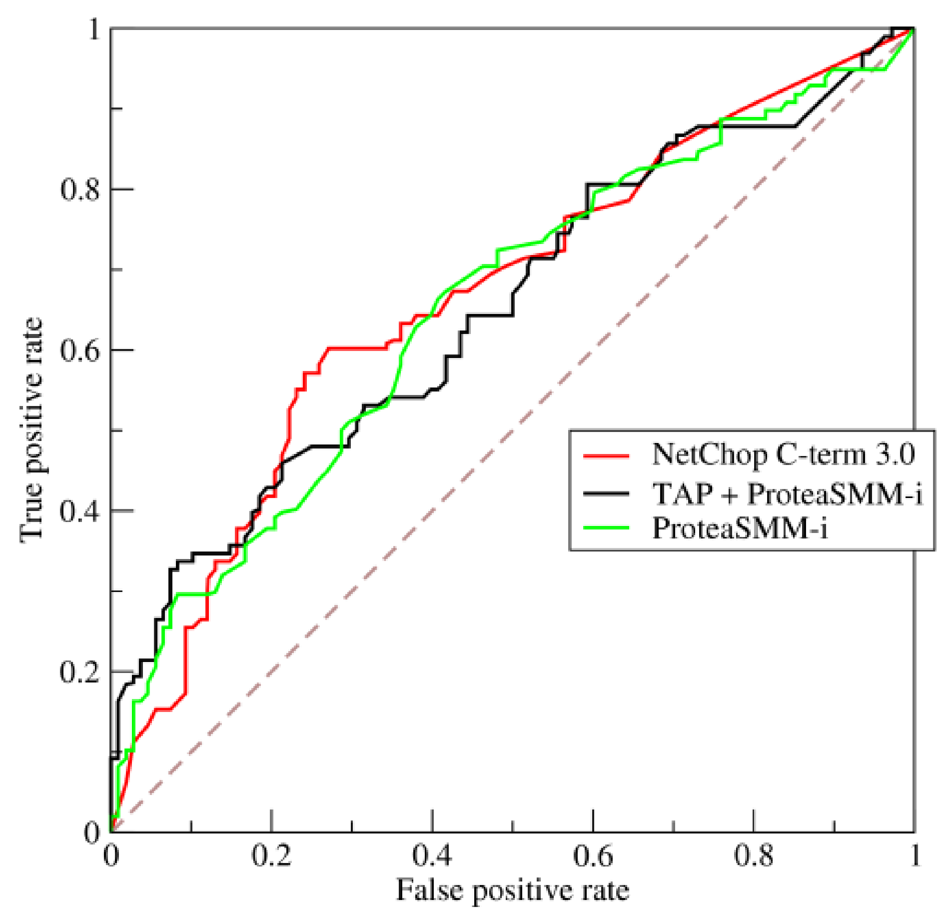

Figure 4: ROC

Is a graphical plot that illustrates the performance of a binary classifier model (can be used for multi class classification as well) at varying threshold values.

Plots the true positive rate against false positive rate.

1.3.1. ROC vs PR

PR doesn't count True Negative at all -> positive class is prioritized implicitly. Precision = TP/(TP + FP) Recall = TP/(TP + FN)

ROC count all. Recall = TP/(TP + FN). FPR = FP/(FP + TN)

In situations where the dataset is highly imbalanced, the ROC curve can give an overly optimistic assessment of the model's performance. This optimism bias arises because the ROC curve's false positive rate (FPR) can become very small when the number of actual negatives is large. As a result, even a large number of false positives would only lead to a small FPR, leading to a potentially high AUC that doesn't reflect the practical reality of using the model.

PR should be preferred if:
- positive class is rare
- false positives is more concern than false negatives

1.4. Multiclass classification

One versus rest strategy - train classifier for each class and pick the class with highest classifier score. One versus one - 1 or 2 classifier, N*(N-1)/2 classifier for classes(good for SVM).

| SGD. All it does is assign a weight per class to each pixel, and when it sees a new image it just sums up the weighted pixel intensities to get a score for each class. So since 3s and 5s differ only by a few pixels, this model will easily confuse them.

1.5. Multilabel classification

Can be done with KNeighbors.

$$\text{F score} = 2 \text{ precision} * \frac{\text{recall}}{\text{precision} + \text{recall}}$$

To evaluate performance of model may use average F_1 score.

1.6. Multioutput classification

Each label can have more than two possible values

1.7. K nearest neighbors

Non-parametric supervised learning method. Used for classification and sometime for regression. In regression output is value for the object - average of the values of k nearest neighbors. In classification output is class for the object - most common class of neighbors.

The algorithm principle relies on distance so normalizing features is crucial. May be used with weighting scheme to prioritize classes of nearer neighbors.

The distance may be Euclidean

Sensitive to skewed class distribution.