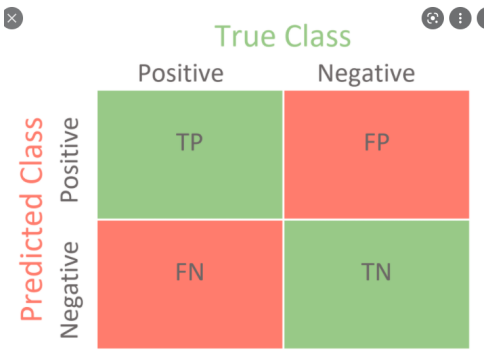
4️⃣

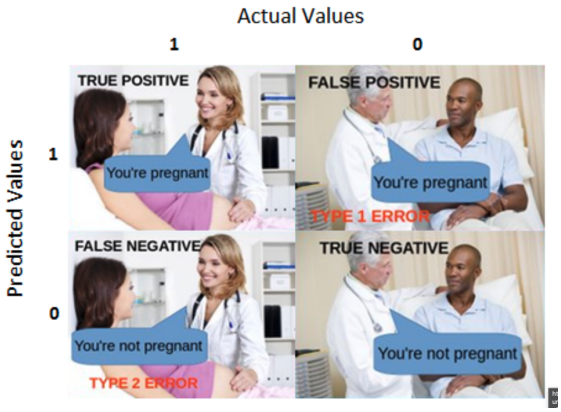
Classification

Status	Done
Files	
Link	
Type	All Around
Course	Data Analysis

Evaluation

Confusion Matrix





Accuracy Recall Precision F1

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{all}}$$

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

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

- Precision and recall are inversely related
- Precision and recall can be combined to make F1

$$\text{F1-measure} = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

- F1 is highest when precision and recall are similar

Logistic Regression

- Makes categorical classifications
- General form:

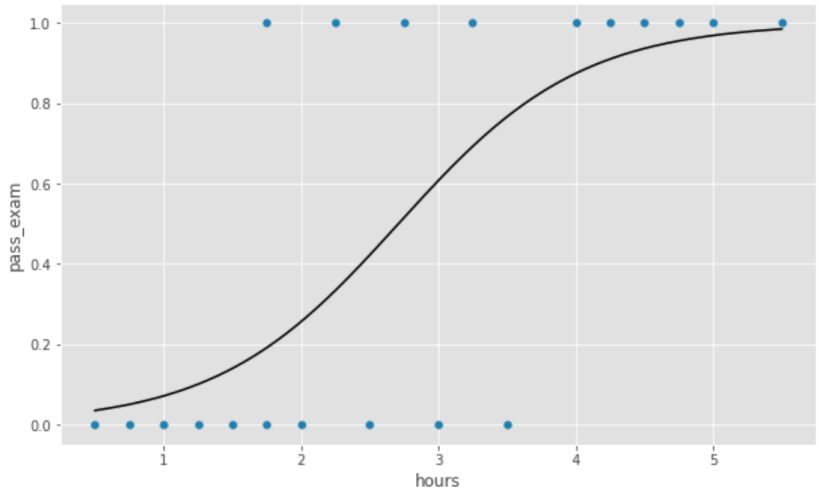
Logistic regression: Given samples (x_i, y_i) for $i = 1, \ldots, n$, find the *best* values of β_0 and β_1 so that

$$y = \text{logistic}(\beta_0 + \beta_1 X) \qquad \text{or} \qquad \text{logit}(y) = \beta_0 + \beta_1 X.$$

- The logistic and logit function are similar

$$\text{logistic}(x) := \frac{e^x}{1 + e^x} = \frac{1}{1 + e^{-x}}, \qquad \text{logit}(p) := \log\left(\frac{p}{1 - p}\right)$$

- What do the coefficients mean?
  - β_0 shifts the curve right or left and β_1 controls how steep the S-shaped curve is.
  - If β_1 is positive, then the predicted $P(Y = 1)$ goes from 0 for small values of X to 1 for large values of X and if β_1 is negative, then $P(Y = 1)$ has the opposite association.
- Example:

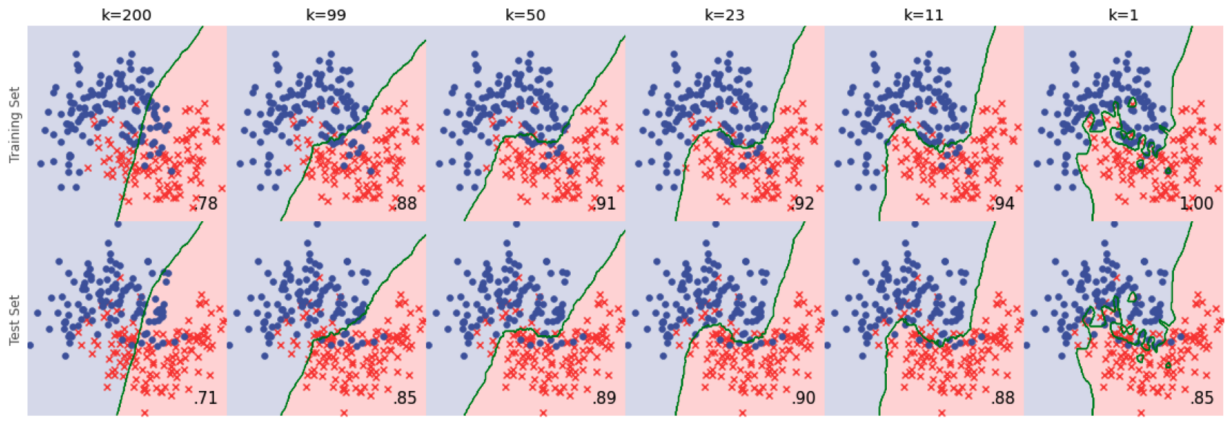


$$p(X) = \text{probability(passing| X hours studying)} = \text{logistic}(\beta_0 + \beta_1 X)$$

- Logistic regression can be generalised to take into account more than one variable and more than one class

k-NN

- k nearest neighbours vote on what class the sample should be
- Need to define which distance measure is used to find neighbours
- Need to pick what k is
  - Low k ⇒ complex wiggly high order polynomial that does not generalise/fit new data very well
  - Hyper-parameter, needs to be checked using a validation set



k-NN vs Logistic

For *good* choices of the parameter k, k-NN has better performance than logistic regression. Logistic regression suffers because the decision boundary isn't curved. For this reason, it is called a *linear classifier*. (However there are extensions to logistic regression that allow the decision boundary to curve).

Decision Trees

- General idea for building them
  - Start with empty tree
  - Choose best predictor and best threshold for splitting
    - Best predictor? → the one that minimises error, error measure?
  - Repeat on each node until stop condition is met

        - Stop condition?
- Advantages
    - Easy to understand and interpret, nice graphical interface
    - Easy to handle categorical (no dummy variables needed)
- Disadvantages
    - Don't have the predictive accuracy of other approaches as they tend to overfit the data.
    - Non-robust, *i.e.*, sensitive to small changes in the data. They demonstrate high variance.

Overfitting

- Big issue in decision trees
- **max_depth:** The maximum depth of the tree. If this is not set, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.
- **min_samples_split:** The minimum number of samples required to split an internal node: If the value is an integer, then consider `min_samples_split` as the minimum number. If it is float, then `min_samples_split` is a percentage and `ceil(min_samples_split * n_samples)` are the minimum number of samples for each split.

Ensemble Methods

- Use multiple algorithms at the same time and then come to a consensus of a predictive label
- Higher accuracy, but also take longer to train

Bagging

- General several models based on subsets of the data, and let these individual models vote to arrive at a prediction

- Averaging these observations reduced variance and this increases the predictive accuracy
- Allows us to increase max_depth, since we rely on the large number of models to reduce variance
- Advantages:
  - high expressiveness - can model complex functions and decision boundaries
  - low variance
  - Out of bag sample for validation, points that were not sampled when building the trees can be used for validation
- Disadvantages:
  - No longer easily interpretable
  - Sometimes can get similar models if certain variables are particularly strong predictors, especially if early on

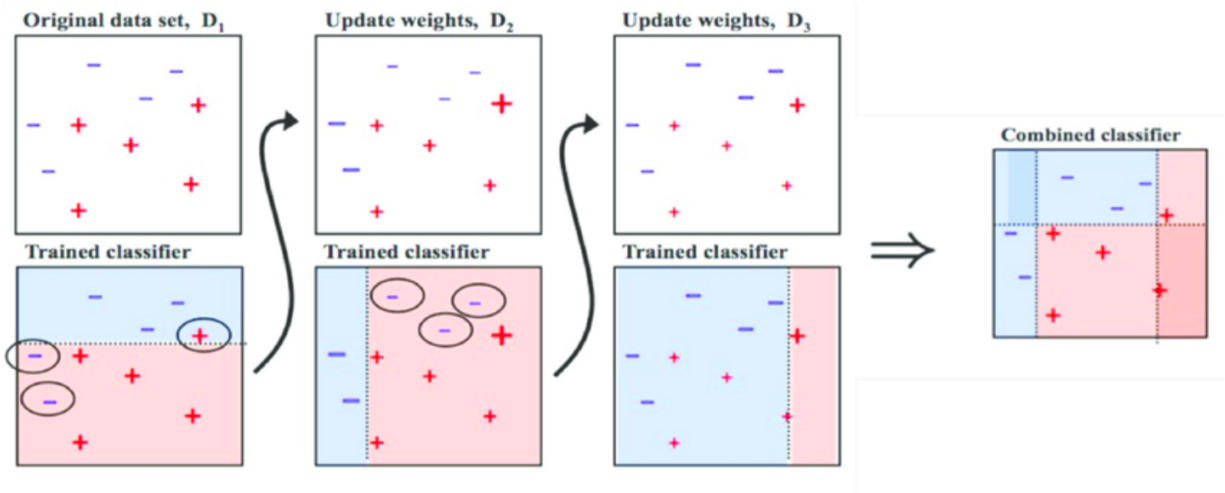
Random Forests

- Similar to bagging, but addresses the limitations of getting similar trees if there are strong predictors
- What variable(s) to split is chosen randomly at each split, hence we get random trees and a random forest
- Hyperparameters
  - Number of predictors m to randomly select at each split, typically $m = \sqrt{p}$ where p is the number of predictors
  - Number of trees
  - Minimum leaf node size / max depth / minimum samples split (same as normal decision trees)
- Perform poorly when number of predictors is big, and the relevant ones are few, because the chances of choosing relevant predictors at splits is small so most of the trees will be weak models

- More trees does not mean overfitting, but if too large might become correlated increasing variance

Boosting

- Learning from mistakes
- General approach that can also be used for other statistical learning methods
- Gradient boosting for regression (models residuals)
- AdaBoost for classification for classification (changes the weights of the data based on misclassifications)



- Parameters
  - Number of trees q, if q is large can encounter overfitting but thats usually slow and rarely happens
  - Learning rate λ, how much is learned from each sample, typically 0.01 or 0.001, small λ needs large q in order to achieve good performance
  - Number of splits d in each tree, which controls the complexity of the boosted ensemble. Often d = 1 works well, in which case each tree is a stump, consisting of a single split and resulting in an additive model

AdaBoost

- Creates a forest of stumps (stumps are trees with one one decision node and two leaves), stumps are weak learners but thats fine for AdaBoost (actually its necessary)
- The stumps (unlike forest for example) dont get equal voting rights, they are weighted
- The order in which stumps are made maters, because the next stump is made based on errors the previous one made
- More here https://www.youtube.com/watch?v=LsK-xG1cLYA on how it works

XGBoost

- Super beast alpha algorithm that can beat anything in the world thing https://www.youtube.com/watch?v=Vly8xGnNiWs