1. Classification vs Regression

Your goal is to identify students who might need early intervention - which type of supervised machine learning problem is this, classification or regression? Why?

Classification is a type of supervised machine learning problem where inputs are classified into discrete outputs. Regression is a problem where inputs are extrapolated to continuous outputs.

The goal of this project is to identify if students will pass or fail their high school exam to identify those students who might need early intervention. Since we are trying to identify if a student will belong to a certain class of being in need of intervention or not. This is a classification problem

2. Exploring the Data

Can you find out the following facts about the dataset?

- *Total number of students: 395*
- *Number of students who passed: 265*
- *Number of students who failed: 130*
- *Number of features: 30*
- *Graduation rate of the class: 67.09%*

3. Preparing the Data

Execute the following steps to prepare the data for modelling, training and testing:

- Feature column(s) are:

  ['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu', 'Fedu', 'Mjob', 'Fjob', 'reason', 'guardian', 'traveltime', 'studytime', 'failures', 'schoolsup', 'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout', 'Dalc', 'Walc', 'health', 'absences']

- Target column is: *['passed']*
- Processed feature columns (43): -

  ['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu', 'Fedu', 'Mjob_at_home', 'Mjob_health', 'Mjob_other', 'Mjob_services', 'Mjob_teacher', 'Fjob_at_home', 'Fjob_health', 'Fjob_other', 'Fjob_services', 'Fjob_teacher', 'reason_course', 'reason_home', 'reason_other', 'reason_reputation', 'guardian_father', 'guardian_mother', 'guardian_other', 'traveltime', 'studytime', 'failures', 'schoolsup', 'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout', 'Dalc', 'Walc', 'health', 'absences']

- Training set: *300 samples* Test set: *95 samples*

4. Training and Evaluating Models

Choose 3 supervised learning models that are available in scikit-learn, and appropriate for this problem. For each model:

Decision Tree Classifier

- What are the general applications of this model?

  Decision tree is a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. It can be used in scenarios where the feature space can be split into rectangles (or in higher dimensions, hyper-rectangles) based on the decision boundaries at each node of the tree.

- What are its strengths and weaknesses?

  The strengths of this model are:
  Simple to understand and to interpret. Trees can be visualised.
  Requires little data preparation or pre-processing..
  The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree. Performs well even if its assumptions are somewhat violated by the true model from which the data were generated.

  The weaknesses of decision trees include:
  Decision-tree learners can create over-complex trees and hence prone to overfitting.
  Decision trees can be unstable because small variations in the data might result in a completely different tree being generated.
  The problem of learning an optimal decision tree is known to be NP-complete under several aspects of optimality and even for simple concepts.
  There are concepts that are hard to learn because decision trees do not express them easily, such as XOR, parity or multiplexer problems.
  Decision tree learners create biased trees if some classes dominate. It is therefore recommended to balance the dataset prior to fitting with the decision tree.

- Given what you know about the data so far, why did you choose this model to apply?

  Decision Tree is easy to visualize and hence easy to understand. The training dataset is small.

```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
        max_features=None, max_leaf_nodes=None, min_samples_leaf=1,
        min_samples_split=2, min_weight_fraction_leaf=0.0,
        presort=False, random_state=None, splitter='best')
```

	Training set size		
	100	200	300
Training time (secs)	0.004	0.008	0.008
Prediction time (secs)	0.001	0.001	0.001
F1 score for training set	1.0	1.0	1.0
F1 score for test set	0.70796460177	0.763358778626	0.741935483871

Support Vector Machine

- What are the general applications of this model?

  SVMs can be used to solve various real world problems:

  Classification of images can be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query methods.
  SVMs are also useful in medical science to classify proteins with up to 90% of the compounds classified correctly.Hand-written characters can be recognized using SVM

- What are its strengths and weaknesses?

  The strengths of support vector machines are:
  Effective in high dimensional spaces.
  Still effective in cases where number of dimensions is greater than the number of samples.
  Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
  Versatile: different Kernel functions can be specified for the decision function.

  The weaknesses of support vector machines include:
  If the number of features is much greater than the number of samples, the method is likely to give poor performances.
  SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

- Given what you know about the data so far, why did you choose this model to apply?

  Since the training set is small but there are many features compared to the size SVM is used.

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape=None, degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)
```

	Training set size		
	100	200	300
Training time (secs)	0.006	0.014	0.050
Prediction time (secs)	0.003	0.005	0.007
F1 score for training set	0.864864864865	0.872131147541	0.876595744681
F1 score for test set	0.783783783784	0.767123287671	0.758620689655

Logistic Regression

- What are the general applications of this model? What are its strengths and weaknesses?

Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function,
Logistic regression is used widely in many fields, including the medical and social sciences
Ex: to predict whether an American voter will vote Democratic or Republican, based on age, income, sex, race, state of residence, votes in previous elections, etc.

Strengths of this model are:
Logistic regression is intrinsically simple, it has low variance and so is less prone to over-fitting.
Works very well for dataset with huge features compared to training size.
It is fast algorithm and hence training and prediction times are very low as seen in the table below

Weakness of this model are:
It is a generalized linear model and hence works if the decision boundary is linear.

- Given what you know about the data so far, why did you choose this model to apply?

The data set is small but with a lot of features 30. The model gives a probability or maximum likelihood for the target to be true. We are trying to identify correctly if a student will pass or not and hence Logistic Regression seems appropriate.

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
      intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
      penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
      verbose=0, warm_start=False)
```

	Training set size		
	100	200	300
Training time (secs)	0.005	0.009	0.017
Prediction time (secs)	0.001	0.001	0.002
F1 score for training set	0.859259259259	0.833922261484	0.839285714286
F1 score for test set	0.751879699248	0.788321167883	0.8

5. Choosing the Best Model

Based on the experiments you performed earlier, in 2-3 paragraphs explain to the board of supervisors what single model you choose as the best model. Which model has the best test F1 score and time efficiency? Which model is generally the most appropriate based on the available data, limited resources, cost, and performance? Please directly compare and contrast the numerical values recorded to make your case.

Logistic Regression is the best model for the given dataset. It has the best test F1 score 0f 0.8 compared to 0.71 and 0.75 of the other two models. It also has the fastest training and prediction time compared to the other two models. Comparing the tables above it is clear that the model has the fastest prediction time. It is able to achieve such a high f1 score even with less training data and hence also used optimum and limited resources for memory and performance. This can help in reducing the cost of running this program.

In 1-3 paragraphs explain to the board of supervisors in layman's terms how the final model chosen is supposed to work (for example if you chose a decision tree or support vector machine, how does it learn to make a prediction).

The logistic regression model works by calculating a maximum likelihood probability for the different outcomes (pass or fail) that can happen based on variables like study time and/or previous failures. The model learns to calculate the coefficients for the logistic function by learning the importance of different features. To make a prediction the model simply plugs in the features of a new student along with the coefficients it has learnt by training to generate an outcome of whether the student will pass or not

What is the model's final F1 score after fine tuning with GridSearch?

*The model was trained for different coefficient values C= 1, 10, 100 using 5 fold CV. The best f1 score of **0.8** was achieved for **C=1***