CQF Exam Three
Implement Choice **A or B**

January 2022 Cohort

Instructions: The submitted report must present work and outputs clearly separated *by Question*. Submit ONE PDF OR HTML file, named LASTNAME_REPORT_E3_**AB**, and ONE zip file named LASTNAME_CODE.zip that includes code, data and any other files. Python notebook with auxiliary output (data, plots) is not an analytical report: such submission will receive a deduction.

Please do not discuss this assignment in groups or messengers. Portal and upload questions to Orinta.Juknaite@fitchlearning.com. Potential discrepancies to Richard.Diamond@fitchlearning.com. Tutors will not provide alternative explanation, discussion or hints beyond exam paper text.

Trend Prediction: Short-term asset return is a challenging quantity to predict. Efficient markets produce near-Normal daily returns with no significant correlation between r_t, r_{t-1}. This exam is a limited exercise in supervised learning: use a set of features from Table 1 without an expectation of high accuracy scores in prediction.

- Choose **one ticker** of your interest form: equity, ETF, crypto token, or commodity.
  Do not choose: FX tickers (GBPUSD), equities with market cap over 70 bln USD.
- Predict **trend only**, for a short-term return (daily, 6 hours). We limit prediction to binomial classification: dependent variable is best labelled $0, 1$ vs. $1, -1$.
  Devise own approach on how to categorise small near-zero returns (drop from training sample, group with positive/negative). The threshold will strongly depend on your ticker.
  Example: small positive returns below 0.25% can be labelled as negative.

Feature	Formula	Description
O-C, H-L	Open - Close, High - Low	of price
Sign	sign $[\ln \frac{P_t}{P_{t-1}}]$, sign $[P_t - P_{t-k}]$	sign of return, sign of momentum
Past Returns	$r_{t-1}, r_{t-2}, \ldots$	shift column of $t-1$ to obtain $t-2$
Momentum	$P_t - P_{t-k}$	price change period k days
Moving Average	$\text{SMA}_i = \frac{1}{n}\sum_{i=0}^{n-1} P_{t-i}$	simple moving average
Exponential MA	$\text{EMA}_t = \text{EMA}_{t-1} + \alpha\,[P_t - \text{EMA}_{t-1}]$	recursive, $\alpha = 2/(N_{obs}+1)$

Table 1: Features to choose from.

Tutors will not discuss the choice of features – eg 3-day vs 5-day Momentum – it is part of exam work to experiment as well as make sense of instructions into implementation. Series length is another decision for you. If predicting short-term trend, up to 5-year period should be sufficient to cover both, train and test periods.

Apply tasks below as appropriate to each model, omissions will lead to the loss of marks.

Choice A. Support Vector Machines

You are likely to produce up to 3 models of type (1), a few features within each, as low as two features per model. Type (2) model has to improve prediction of negative moves.

Question 1. First, theory tasks:

- Present the maths of SVM classifier: specifically, hyperplane and margin equations.
- If you plot 'support vectors' what would they be? (the output obtained with *SVM_SVC.supportvectors*)

Question 2. Implementation tasks for models type (1), (2) as appropriate.

- Make a decision about *StandardScaler* and the regressor vs classifier version of SVM.
- Present comparisons of soft vs. hard margin and draw conclusions on which 'works better'.
- Provide 2D illustrations of linear hyperplanes (feature vs. dependent variable) – two per a model type (1) and two per (2). Provide one 2D illustration for a deliberate data leakage, eg when dependent variable (return) and momentum are computed using the same P_t input.
- Investigate the prediction quality using area under ROC curve (each class) and confusion matrix. Do report on results for train/test split (50%/50% of observations is a common split).

Question 3. SVM is a very generalisable technique – if you manage to make hyperplane separation work for your data (i.e., achieve a linear separation in 2D relationship of feature vs dependent variable(s)). SVM prediction is improved by feature scaling (see Python Lab on regression method) and use of kernel transformation. For the latter, Radial Basis Function is a common kernel, if you utilise it you have to explain the role of *gamma* kernel parameter.

Do not present all attempted scaling/kernels but write a brief empirical result of (a) feature scaling and (b) impact of kernel on SVM classification results – three paragraphs minimum.

Question 4. Provide P&L backtesting plots (cumulative returns) for Kelly optimal bets vs. 100% bets. Kelly allocation to the asset computed with probability p taken from *predict_proba()* is as follows:

$$p - (1 - p) = 2p - 1$$

Therefore, you remain in cash for $1 - (2p - 1) = 2(1 - p)$ percentage, which reduces your gain (and loss). *Worked example:* allocation 75% into a risky asset that moves up 5%, while 25% remained in a broker account earning 0%, gives the total gain of $0.75 \times 0.05 + 0.25 \times 0 = 3.75\%$.

To complete this section, use *predict_proba()* output from any one model – type (2) recommended. Assume daily betting on price direction. Use the realised return value with PREDICTED sign to compute profit (loss) for end of the day. P&L can take a more profitable path than the asset price evolution (long-only strategy) because of successful bets on negative moves.

Choice B. Decision Trees and Ensembles

While trees suffer from high dimensionality too, it is possible to work with more than a few features per model. Produce two models: (1) initial mapping model with many features, and (2) reduced in features model that aims to improve prediction of negative moves as well as (3) study design for a boosted model.

Question 1. First, theory tasks:

- Mathematically specify three types of Loss Function for decision trees.
- Which data transformation can prepare the data for linear separation? (answer to score marks)

Question 2. Implementation tasks for models type (1), (2) as appropriate.

- Vary hyperparameters (min number to split, minimum number in leaf, and maximum depth) – can use ready search *RandomizedSearchCV*, *GridSearchCV*. While you are not required to generate surfaces, please.
- Formulate up to four principles/purposes of trees pruning. Illustrate with model type (2) vs (1) as appropriate.
- Provide attribution to splits (discuss if they are sensible), particularly for the model type (2) better **negative moves** classification.
- Provide plots for decision boundaries (surfaces), and name two specific issues for prediction quality of tree models. Investigate the prediction quality using area under ROC curve (each class) and confusion matrix. Do report on results for train/test split (typically, 50%/50% of observations).

Section 3. Ensemble learning

- Experiment whether to use Adaboost or XGBoost, with which inputs: tree regressor or classifier, which specific tuned hyperparameters, pruned or non-pruned. Do not present all runs but write a brief empirical study design for (3) a boosted model of your choice – three paragraphs of text minimum.

Question 4. Provide P&L backtesting plots (cumulative returns) for Kelly optimal bets vs. 100% bets. Kelly allocation to the asset computed with probability p taken from *predict_proba()* is as follows:

$$p - (1 - p) = 2p - 1$$

Therefore, you remain in cash for $1 - (2p - 1) = 2(1 - p)$ percentage, which reduces your gain (and loss). *Worked example:* allocation 75% into a risky asset that moves up 5%, while 25% remained in a broker account earning 0%, gives the total gain of $0.75 \times 0.05 + 0.25 \times 0 = 3.75\%$.

To obtain sensible probabilities from trees method you might need to use Regressor or non-optimal min_samples_leaf. Assume daily betting on price direction. Use the realised return value with PREDICTED sign to compute profit (loss) for end of the day. P&L can have a more profitable path than the asset price evolution (long-only strategy) because of successful bets on negative moves.

END OF EXAM THREE TASKS