

Домашняя работа 3

Для выполнения задания ДЗ 3 использовать базу данных, созданную в результате ДЗ 2.

Вариант 1. Кредитные рейтинги

1. Для заданного пользователем **периода** и **типа рейтинга** построить распределения допустивших и недопустивших дефолт заемщиков по рейтинговым категориям. Формат результата: таблица (DataFrame) с полями **rating**, **non-default**, **default**, отсортированная в порядке улучшения рейтинга. В таблице должны быть учтены все рейтинговые категории (корме, Снят, Приостановлен), даже если наблюдений с таким рейтингом нет. При выполнении пункта использовать данные и шкалу агентства EXP (ЭкспертРА). (1 балл)
2. Визуализировать распределения с помощью графиков вида **bar (не stacked)**. Для распределений дефолтеров и недефолтеров использовать разные цвета. Подписи на оси x должны соответствовать символам рейтинговых категорий в порядке ухудшения кредитного качества. График и оси должен быть подписаны. На графике должна быть легенда. При выполнении пункта использовать данные и шкалу агентства EXP (ЭкспертРА). (2 балл)
3. Используя таблицу из пункта 1, вычислить эмпирические функции распределения недефолтеров и дефолтеров по рейтинговым категориям (sumsum столбца исходной таблицы/sum соответствующего столбца). По этим данным построить ROC-кривую с помощью графика plot с любым маркером (x – значение функции распределения для недефолтеров, y – значение функции распределения для дефолтеров) (см. II.2. Receiver Operating Characteristic на стр. 11: <https://www.bundesbank.de/resource/blob/704150/b9fa10a16dfff3c98842581253f6d141/mL/2003-10-01-dkp-01-data.pdf>). Подписать оси. При выполнении пункта использовать данные и шкалу агентства EXP (ЭкспертРА). (2 балл)

Вариант 2. Облигации

1. Для заданного пользователем **эмитента** и **даты** построить график платежей по портфелю, содержащий по одной из всех облигаций эмитента из пункта 3 из ДЗ 2. Формат результата: таблица (DataFrame) с полями **date**, **zero-coupon**, **plain_coupin**, **amortized** отсортированная в порядке возрастания даты. (1 балл)
2. Визуализировать поток платежей с помощью графика вида **stacked bar**, используя разные цвета для облигаций разного типа (предусмотренные в пункте 3 ДЗ 2). По оси x указать даты выплат. В легенде указать соответствие цветов и типов облигаций. В названии объяснить содержимое графика, а также указать заданные пользователем эмитента и дату. (2 балл)
3. Используя данные столбцов [Y2O_ASK, Y2O_BID, YIELD_ASK, YIELD_BID на выбор] таблицы quotes на заданную дату, вычислить дюрацию (коэффициент чувствительности стоимости облигации к изменению ее доходности) (см. формулу (4.9) на стр. 6: https://www.csie.ntu.edu.tw/~lyuu/Capitals/lessons_dur.pdf) для каждой облигации в портфеле. Если значение в соответствующем поле отсутствует, вычисления не производить. Результаты визуализировать с помощью графика типа **scatter** в осях (x – ISIN, y – дюрация). Значения должны быть упорядочены в порядке возрастания дюрации. Подписать оси. В названии указать количество облигаций, для которых не удалось рассчитать дюрацию. (2 балла)

Материалы для сдачи

1. Краткая инструкция, содержащая:
 - а. подготовительные действия для работы с данными (где и в каком виде хранить данные, какой софт установить и т.д.);
 - б. краткое содержания процедур, заложенных в скриптах/запросах, а также порядок их запуска;
 - с. описание ожидаемых результатов каждого этапа.
2. Код сдается в виде одного *.ipynb с комментариями и выполненными ячейками.
3. Все (кроме данных) необходимо разместить в репозиторий на github, в котором преподаватель является участником с правом редактирования.

Требования к выполнению задания

1. Все процедуры и результаты должны быть воспроизводимы и описаны в мере, обеспечивающей использование третьими лицами.
2. Все пользовательские параметры, в том числе, параметры подключения к базе данных должны быть обособлены в отдельные переменные в начале скрипта.

Правила и сроки выполнения

1. Домашняя работа является индивидуальной активностью студента.
2. Срок выполнения (первая итерация сдачи) Домашней работы 3: **20 декабря 23:59:59**.
3. Максимальная оценка за выполненное задание: **5 баллов**.
4. По результатам первой итерации студент получает фидбэк и предварительную оценку. Ориентировочно через **3 дня после сдачи работы**. Получив фидбэк, студент может исправить работу и сдать ее до **23 декабря**, но не позднее, чем последующие домашние работы по курсу.
5. Студент пропустивший дедлайн первой итерации сдает работу на итоговую проверку **без возможности ее исправить**.
6. Студент выполнивший все пункты Домашней работы 3 (не зависимо от итерации) получает право проверки аналогичной работы сокурсника. Срок проведения peer-review: **2 дня с момента получения задания на проверку**. Продление сроков peer-review допускается по договоренности с преподавателем.
7. Peer-review проводится по следующим аспектам:
 - *Полнота* (по всем ли пунктам была предпринята попытка выполнения?)
 - *Понятность* (инструкция понятно написано? понятно ли прописаны требования к исходным данным? понятно ли, что делает код и каким результатам это приводит? пользовательские параметры обособлены и понятны?)
 - *Содержание* (сделано именно то, что требуется в задании? если нет, то в чем отличия того, что сделано? в чем это выражается и что Вы сделали, чтобы это увидеть?)
 - *Корректность* (разумно ли выбраны форматы поле таблиц? присутствуют ли ошибки исполнения кода или смысловые ошибки? в чем именно ошибка? как Вы это обнаружили? знаете ли Вы как это исправить? видите ли Вы заведомо абсурдные результаты?)
8. Максимальная оценка за peer-review: **4 балла**.