**Документация, содержащая описание функциональных характеристик экземпляра программного обеспечения**

**г. Москва 2024**

**Содержание**

[1 Предназначение ПО 3](#_heading=h.s2xpnm9x5zoo)

[2 Описание функциональных характеристик 3](#_heading=h.7ep7zxhbxr7i)

[3 Программные и аппаратные требования 4](#_heading=h.fuxsqccpbuer)

[4 Состав системы 5](#_heading=h.9nne77bsuqky)

[5 Эксплуатация системы](#_heading=h.s1q4gko3eb1g) 8

[5.1 Запуск системы](#_heading=h.pl5dbvabgvx9) 8

[5.2Управление](#_heading=h.x5x87cvvpbqe) 8

[5.3 Резервное копирование и восстановление](#_heading=h.2kvj0xbo36g8) 9

[5.4 Обновление](#_heading=h.pc23rfs11dej) 9

[5.5 Техническая поддержка](#_heading=h.p06wm69d70nt) 9

[5.6 Завершение работы](#_heading=h.21s043gw3ecm) 9

# **1 Предназначение ПО**

Python3 распространяемая библиотека fdwscoring (Fordewind Scoring) является программным обеспечением, предназначенным для оценки кредитоспособности юридических лиц малого и среднего бизнеса и индивидуальных предпринимателей. Данное решение позволяет проводить многосторонний скоринг заемщиков на основе финансовых показателей бухгалтерского учета и данных коммерческой бизнес деятельности.

# **2 Описание функциональных характеристик**

1. Оценка кредитоспособности юридических лиц и индивидуальных предпринимателей. Библиотека содержит различные классы моделей, которые могут быть использованы для решения широкого круга прикладных задач. Модели используют семантически разные данные, включая финансовые показатели заемщика, информацию об арбитражных делах и исполнительных производствах, данные о гос. закупках и прочее.

2. Проведение скоринга надежности заемщика. Библиотека содержит триплет схожих моделей, каждая из которых позволяет оценить надежность заемщика простой числовой метрикой от 0 до 1. Данные модели инкапсулируют многолетний опыт и актуальную практику принятия решений финансовых аналитиков компании Fordewind. Функционал ПО позволяет получить score и итоговое решение о статистически значимом наличии или о статистически значимом отсутствии кредитной надежности заемщика.

3. Оценка вероятности дефолта юридического лица. Библиотека содержит модель, которая позволяет оценивать вероятность дефолта юридического лица. Данная модель построена на объективных данных по дефолтам юридических лиц. Функционал ПО позволяет получить оценку вероятности дефолта и статистически значимое решение по дефолту. Последнее подразумевает, что в случае, если оценка вероятности дефолта выше внутренне установленного порогового значения, то модель в качестве решения выдает потенциально возможный дефолт; в противном случае, решение модели не является статистически значимым.

4. Поиск и анализ риск-факторов заемщика. Библиотека содержит две модели для выявления риск-факторов юридических лиц и индивидуальных предпринимателей, соответственно. Функционал ПО позволяет получить список обезличенных риск-факторов, которые с точки зрения уровня риска приблизительно равноценны. Также, исходя из выявленных риск-факторов, модели принимают итоговое решение по заемщику.

5. Оценка лимитов выдаваемых заемщику займов. Библиотека содержит модель, которая отражает актуальную практику финансовых аналитиков компании Fordewind по расчету лимитов займов при работе с компаниями малого и среднего бизнеса. Функционал модели позволяет рассчитать три значения лимита: мало рисковое значение лимита, значение комфортного лимита и высоко рисковое значение лимита.

# **3 Программные и аппаратные требования**

***Программные требования:***

Операционная система. Python3 распространяемая библиотека fdwscoring совместима с операционными системами Windows, Linux и MacOS.

Использование ПО подразумевает установку Python версии 3.8.

Система контроля версий Git. Для получения копии и дальнейшего обновления компонентов ПО, необходимо наличие системы контроля версий Git.

***Аппаратные требования:***

Аппаратные требования для использования ПО продиктованы аппаратными требованиями установки интерпретатора Python3.8.

Процессор. Любой современный процессор с тактовой частотой не менее 1.0 ГГц. Для улучшения производительности рекомендуется использование многоядерного процессора с тактовой частотой не менее 2.0 ГГц.

Оперативная память. Минимальный объем оперативной памяти (RAM) составляет 2 ГБ. Рекомендованный объем оперативной памяти не менее 4 ГБ.

Хранилище данных. Объем жесткого диска зависит от ожидаемого объема данных. Для установки компонентов ПО и непосредственного скоринга одного заемщика, минимальный требуемый объем свободного пространства составляет 200 МБ. Для комплексного анализа 1000 заемщиков рекомендуется иметь не менее 500 МБ свободного пространства.

# **4 Состав системы**

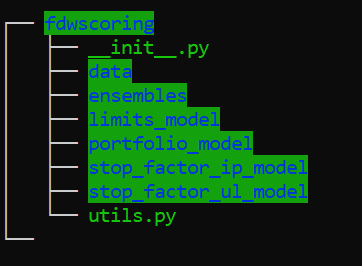
Описание технической инфраструктуры

Язык программирования — Python3.8

Используемая платформа разработки — изолированное виртуальное окружение (virtual environment) Python

Используемые сторонние компоненты ПО – open-source python3 распространяемые библиотеки: "pyyaml", "joblib ", "transliterate", "numpy", "pandas", "scikit-learn", "lightgbm", "catboost".

*Общая структура библиотеки*



data – Каталог, содержащий шаблоны данных по заемщикам. Указанные шаблоны могут быть использованы пользователем для самостоятельной подготовки данных, необходимых для проведения скоринга.

ensembles – Каталог, содержащий в себе триплет ансамблевых моделей, функционал которых позволяет проводить оценку надежности заемщика.

limits\_model – Содержит модель расчета лимитов выдаваемых займов.

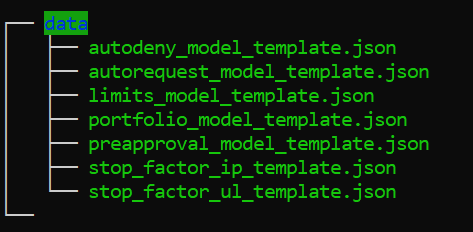
portfolio\_model – Каталог, инкапсулирующий модель оценки вероятности дефолта юридических лиц.

stop\_factor\_ip\_model – Содержит модель поиска и анализа риск-факторов индивидуальных предпринимателей.

stop\_factor\_ul\_model – Содержит модель поиска и анализа риск-факторов юридических лиц.

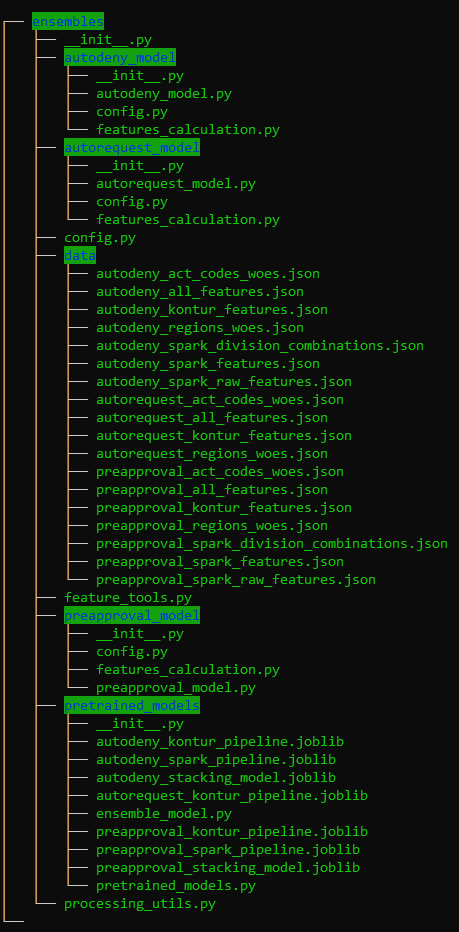
utils.py – Модуль, содержащий вспомогательный функционал.

*Структура каталога data*



Каталог содержит шаблоны данных по заемщикам для каждой модели.

*Структура каталога ensembles*

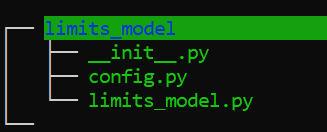


Каталоги autodeny\_model, autorequest\_model и preapproval\_model инкапсулируют AutodenyModel, AutorequestModel и PreApprovalModel модели, соответственно. В указанных каталогах находятся модули autodeny\_model.py, autorequest\_model.py и preapproval\_model.py, соответственно, которые содержат в себе классы верхнеуровневого API. Также каждый каталог содержит вспомогательный модуль features\_calculation.py расчета фичей и модуль внутренней настройки и конфигураций config.py (для каждой модели указанные пары модулей свои).

Каталог data содержит файлы, используемые при работе с данными и при проведении скоринга. Каталог pretrained\_models состоит из .joblib файлов предобученных моделей и вспомогательного ensemble\_model.py модуля.

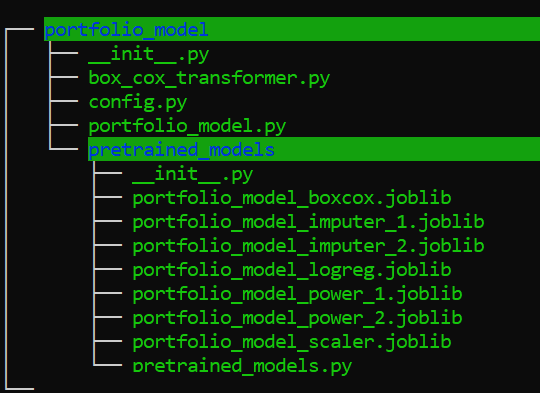
Верхнеуровневые модули feature\_tools.py, processing\_utils.py и config.py используются всеми тремя моделями. Первый модуль содержит функционал расчета фичей, второй модуль содержит функционал обработки сырых данных, третий модуль содержит общие конфигурации моделей.

*Структура каталога limits\_model*



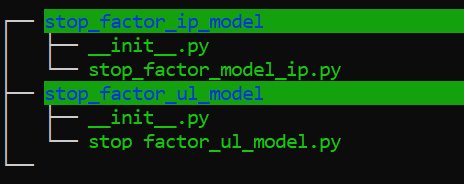
Данный каталог состоит из модуля limits\_model.py, который содержит класс верхнеуровневого API, и config.py, который настраивает и конфигурирует модель.

*Структура каталога portfolio\_model*



Каталог pretrained\_models состоит из .joblib файлов предобученных моделей и вспомогательного pretrained\_models.py модуля. Модуль box\_cox\_transformer.py содержит оригинальную расширенную имплементацию преобразования Бокса-Кокса, применяемого для нормализации фичей. В модуле portfolio\_model.py находится класс модели PortfolioModel верхнеуровневого API. Модуль config.py содержит общие настройки и конфигурацию модели.

*Структура каталогов stop\_factor\_ip\_model и stop\_factor\_ul\_model*



Каталоги имеют схожую структуру и состоят из одноименных модулей stop\_factor\_model\_ip.py и stop\_factor\_model\_ul.py, которые содержат классы StopFactorIPModel и StopFactorULModel верхнеуровневого API, соответвственно.

# **5 Эксплуатация системы**

# 5.1 Запуск системы

Перед запуском системы убедитесь, что все необходимое оборудование находится в рабочем состоянии. Запускайте систему в соответствии с инструкциями, предоставленными в руководстве пользователя.

# 5.2 Управление

Осуществляйте управление системой в рамках импортируемой python3 распространяемой библиотекой. При эксплуатации системы используйте верхнеуровневый API, представленный в руководстве по эксплуатации экземпляра ПО. Перед проведением скоринга заемщиков подготовьте все необходимые данные согласно указанным в составе системы шаблонам (см. п.4), или получите необходимые данные у Заказчика в рамках дополнительной отдельно оплачиваемой услуги.

# 5.3 Резервное копирование и восстановление

В ходе эксплуатации ПО структура и внутренние данные не мутируются, поэтому резервное копирование системы необязательно. В случае утраты экземпляра ПО, пользователи могут обратиться по каналам тех. поддержки для получения копии ПО. Рекомендуется, однако, проводить регулярное резервное копирование пользовательских данных, полученных перед и в ходе эксплуатации системы.

# 5.4 Обновление

Следите за выпуском обновлений для библиотеки fdwscoring и регулярно производите их установку согласно руководству по установке экземпляра  ПО. Осуществляйте техническое обслуживание оборудования пользователя, чтобы предотвратить возможные сбои и проблемы.

# 5.5 Техническая поддержка

Пользователи могут подавать заявки через электронную почту. При составлении заявки необходимо фиксировать основные данные о проблеме, включая описание проблемы, контактные данные пользователя, срочность запроса и другую необходимую информацию.

# 5.6 Завершение работы

Так как ПО представляет собой python3 распространяемую библиотеку, работа с системой завершается автоматически при завершении сессии интерпретатора Python или при прерывании работы ядра интерактивной среды разработки, такой как, например, Jupyter Notebook.