Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный университет пищевых производств»

Кафедра *Автоматизированные системы управления биотехнологическими процессами*

**ОТЧЕТ**

**по производственной практике**

Место прохождения практики ФГБОУ ВО «РОСБИОТЕХ»

Срок прохождения практики с «7» февраля 2023г. по «11» апреля 2023г.

Руководитель практики от выпускающей кафедры Зуева Юлия Викторовна, доцент кафедры АСУБП

*(ФИО, должность)*

Руководитель практики от организации Зуева Юлия Викторовна, доцент кафедры АСУБП

*(ФИО, должность)*

Выполнил:

студент 4-го курса группы 19-УТС-1

института промышленной инженерии, информационных технологий и мехатроники

Корнюхин Даниил Алексеевич/\_\_\_\_\_\_\_\_\_\_\_

*(ФИО, подпись)*

«16» апреля 2023г.

Москва, 2023

Оглавление

[1. Введение 2](#_Toc134885084)

[2. Влияние климатических факторов на виноград 4](#_Toc134885085)

[3. Откуда берем погодные данные 7](#_Toc134885086)

[4. Влияние почвенных параметров 9](#_Toc134885087)

[5. Откуда берутся почвенные данные 13](#_Toc134885088)

[6. Функционал приложения 16](#_Toc134885089)

[6.1 Фронтенд 17](#_Toc134885090)

[6.2 Бэкенд 19](#_Toc134885091)

[6.2.1 Взаимодействие администратора с приложением 20](#_Toc134885092)

[6.2.2 Взаимодействие ученого с приложением 21](#_Toc134885093)

[7. Машинное обучение 21](#_Toc134885094)

[7.1 Преобразование данных для обучение 25](#_Toc134885095)

[7.2 Обучение 27](#_Toc134885096)

[7.3 Проверка обученных данных 30](#_Toc134885097)

[7.4 Получение предсказание 36](#_Toc134885098)

[7.4.1 Фичи 37](#_Toc134885099)

# 1. Введение

**О приложение**

В приложение будет отображаться обычная карта, в ней будет выделен район в котором будет множество точек с различными природными характеристиками:

Погодные параметры:

температура(макс, мин, средняя) за 2 недели,

ветреность,

облачность,

давление,

влажность,

тип погоды (дождь, облачно, солнечно) какие чаще встречаются

свойства почвы:

Объемная плотность,

Катионный обмен,

Крупные фрагменты,

Нитрогены,

Ph,

Концентрация органического углерода,

Гравиметрическое содержание песка, ила и глины

Также будет отображаться коэффициент, который будет отображать схожесть, по средним параметрам, с другими районами где уже достаточно долго выращивают виноград

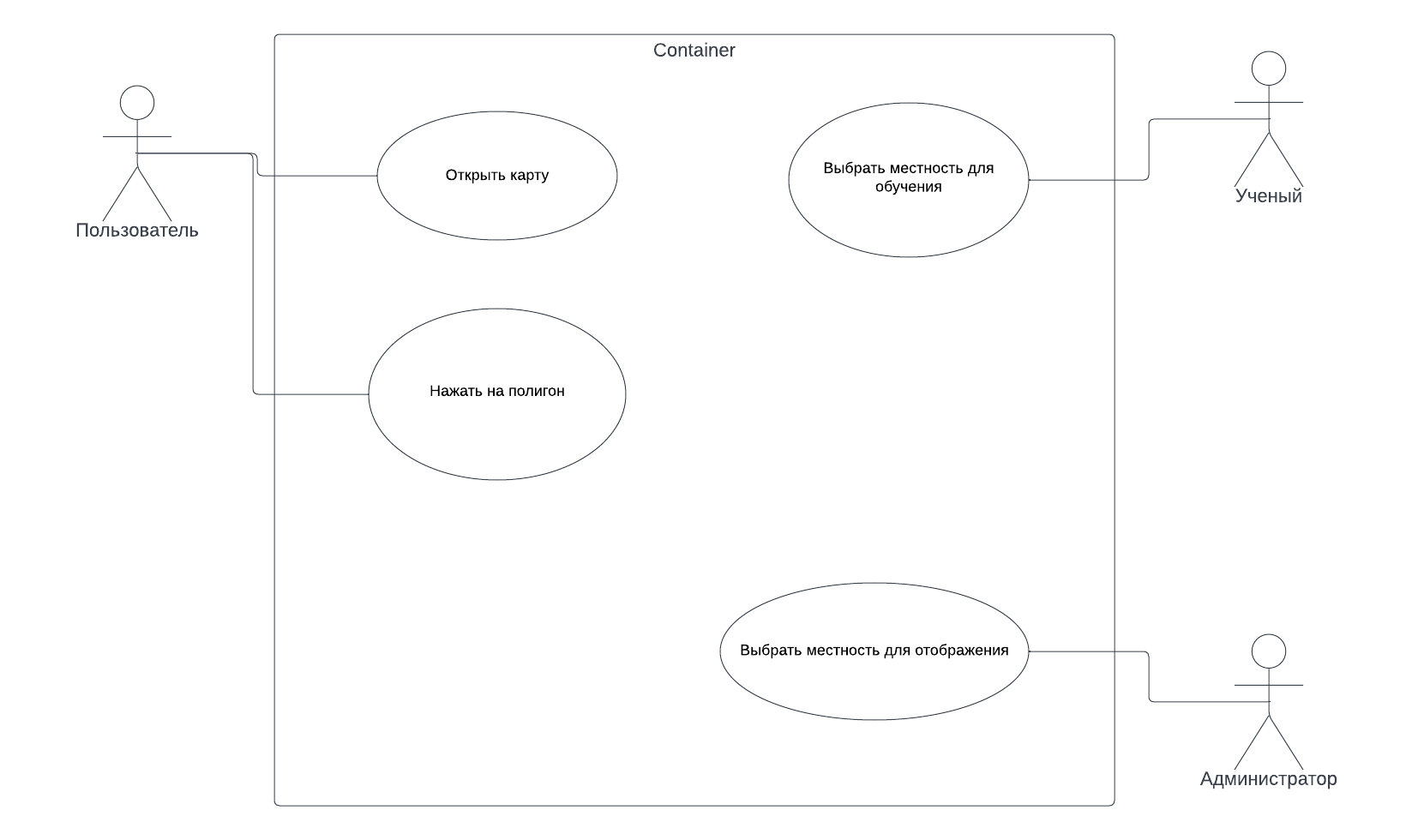
Приложение поможет людям, которые хотят заниматься выращивание винограда, найти наилучшие места. Также если хорошие места будут обладать рядом других недостатков, не связанные с виноградом, (проблема с логистикой и т.д.), можно будет найти места похуже и улучшить их вложившись туда. Например: на потенциальном поле для винограда, растут деревья, тогда достаточно будет вырубить их. Или на потенциальном поле для винограда, в почве будет мало песка, тогда нужно будет добавить побольше песка.

.Им не нужно будет лично разъезжать по разным местам и проверять местность (почва, погода и т.д.), к тому же они могут приехать в удачный день, а в остальные дни там может быть плохая погода и наоборот. Пользователи смогут дистанционно быстро выбрать потенциально хорошие районы и сосредоточится на дальнейшем развитии.

Не нужно будет лично разъезжать по разным местам и проверять местность (почва, погода и т.д.). Пользователи смогут дистанционно быстро выбрать потенциально хорошие районы и сосредоточится на дальнейшем развитии. Например человек может приехать в удачный день, а в остальные дни там может быть плохая погода и наоборот.

В будущем планируется добавить возможность, клика на любую часть карты и получение по этим координатами, природных характеристик и коэффициент.

**UseCase**



**Полигон**

Представляет из себя цветной квадрат, который содержат погодные и почвенные параметры, а также его качество. Параметры берутся от центра квадрата!

# 2. Влияние климатических факторов на виноград

**Температура**

Температура природной среды, амплитуда её колебаний критически влияет на ареал возделывания винограда — она определяет ареал возделывания этой культуры. Поскольку виноград принадлежит к растениям умеренно теплого климата, жаркое лето и относительно теплая зима при прочих благоприятных условиях увеличивают шанс на высокий урожай и положительно влияют на его качество. Существенное значение для виноградной лозы имеет сумма активных температур. Среднесуточная температуре 19—20°С обеспечивает лучшую ассимиляцию диоксида углерода листьям, максимальное накопление сахара в ягодах, уменьшает кислотность виноградного сока. Оптимальными являются температуры (28—32°С) в фазе созревания. При хорошей инсоляции и не особенно большой влажности такие температуры благоприятствуют тому, что в ягодах винограда накапливаются красящие и ароматические вещества. Вина при этом получаются с относительно высокой спиртуозностью и экстрактивностью.

Для южных районов виноградарства обычно характерны вина более полные, ароматичные, тяжелые; для северных районов — более легкие, слабоокрашенные и слабоароматичные.

Но среднесуточная температурa свыше 35°С неблагоприятна для обмена веществ винограда, поскольку она уменьшает ассимиляцию диоксида углеродарастением. От этого созревание ягод задерживается их сахаристость падает, а кислотность их сока растёт.

При ещё более высокой среднесуточной температуре выше — 41—42°С и слабой транспирации виноградарь рискует тем, что листья получат ожоги, кожица ягод морщится; если это происходит в дождливую погоду, то на гроздях способна развиться серая гниль.

**Ветровой режим**

Ветер, как правило, вызывает изменение температуры, влажности воздуха, увеличивает приток С02 к листьям. При этом влажный и умеренный по силе ветер создает благоприятные условия для роста и развития всех органов и формирования качественного урожая ягод винограда.

Ветры могут быть постоянными или возникать в отдельные периоды. Постоянные ветры, продолжающиеся в течение нескольких недель, бывают в Краснодарском крае. Они, как правило, холодные и вызывают снижение температуры воздуха. Постоянные горячие сухие ветры дуют со стороны Терско-Кумского песчаного массива. Они вызывают значительное повышение температуры и резкое снижение относительной влажности воздуха, что ухудшает протекание основных физиологических процессов в растениях и отрицательно сказывается на продуктивности насаждений. Кроме того, с ветром переносятся песчаники, вызывая засекание прироста у молодых посадок и даже их гибель.

Вместе с тем ветер улучшает аэрацию виноградника и таким образом задерживает развитие грибных болезней. Морской влажный ветер способствует лучшему наливу ягод в период созревания винограда, а сухой повышает их сахаристость.

Однако движение воздуха с большой скоростью оказывает отрицательное физическое и механическое влияние на виноградное растение. Он обламывает молодые побеги, иссушает почву и усиливает транспирацию листьев. Особенно сильные ветры могут повредить шпалерную систему на винограднике. Зимой из-за сильных ветров кусты могут засохнуть. Кроме того, ветер, дующий в зимнее время, усиливает отрицательное воздействие на лозу низких температур.

Ослабить вредоносное действие ветра на виноградное растение можно путем размещения насаждений в защищенных от ветра микрорайонах и на склонах, а также посадки защитных лесополос, правильного выбора направления рядов, задернения почвы в междурядьях и др. Эти меры смягчают микроклиматические условия. Виноградники, расположенные вблизи лесных массивов, также защищены от сильных ветров. Вблизи леса климат мягче, поэтому и качество винограда здесь лучше.

**Освещение**

Критическим для винограда является и уровень освещённости. Виноград — светолюбив. Качество и технологических свойства его урожая в большой степени определяются продолжительностью, интенсивностью и спектральными характеристиками освещенности.

При этом, ультрафиолетовая часть спектра влияет на рост, плодоношение, количество гроздей, окраску ягод, биохимические свойства их сока, а красно-желтая часть спектра определяет интенсивность фотосинтеза.

Солнечный свет стимулирует ягоды винограда накапливать красящие вещества. При достаточном солнечном освещении лоза вырабатывает больше углеводов. Это обеспечивает более высокую сахаристость ягод, уменьшает их кислотность и усиливает аромат и окраску.

Кожица ягод на интенсивном свету утолщается. Винограда белых сортов при этом может окраситься в темно-желтый, розовый или бурый цвет.

 Благодаря свету и солнечной энергии в зеленых, листьях образуются органические вещества, нагреваются, почва, растения и воздух, происходит развитие, рост, плодоношение и все те процессы, которые протекают в живом растительном, организме.

Однако свет нужен растениям не только как источник тепловой энергии. Известно, что виноград относится к светолюбивым растениям. Так, например, соцветия, находящиеся в тени, характеризуются плохим оплодотворением и завязыванием ягод.

При умеренной температуре свет оказывает положительное влияние на качество плодов. Грозди, лучше освещенные, имеют более красивую окраску и дают лучшую по качеству продукцию. Вот почему при культуре винограда многие агротехнические приемы направлены на усиление действия прямого солнечного света.

В горных и предгорных районах практически усиление действия света на виноград достигается выбором склонов. На крутые южные, юго-восточные и юго-западные склоны света падает больше, чем на ровные участки или, склоны других экспозиций. Так, например, при крутизне склона в 20—25° света падает на 20—25% больше по сравнению с ровным участком. На склонах северных, северо-восточных и северо-западных соответственно с падением уклона уменьшается освещение.

Однако различные сорта винограда не одинаково относятся к прямому действию солнечного света. Поэтому в соответствии с их отношением к свету можно подбирать и использовать различные склоны. Например, такие сорта, как Рислинг, Сильванер, Шасла, которые издавна культивируются в более северных границах, лучше удаются в южных районах на северных склонах.

Световая стадия развития виноградного растения обеспечивается как при относительно коротком дне (на юге), так и при длинном дне (на севере). Но в этом отношении различные сорта не одинаково чувствительны к силе действия солнечного света и его продолжительности. Как показали исследования, сорта Шасла, Рислинг, Сильванер, Мадлен Анжевин и другие дают больший урожай при более рассеянном свете и при более длинном дне, т, е. эти сорта повышают свою продуктивность до мере продвижения на север.

Лучшее освещение всех органов виноградного растения достигается необходимой густотой посадки, созданием соответствующей формы кустов, удалением бесплодных побегов и пасынков, своевременной подвязкой однолетних побегов, правильным их распределением в пространстве и другими приемами.

Учитывая, что абсолютное большинство сортов требует сильного освещения, улучшая при этом качество продукции, вся забота виноградарей при уходе за насаждениями должна быть направлена к созданию наилучшего освещения всех органов виноградного растения, особенно при культуре винограда в более северных районах СССР.

Как уже отмечалось выше, комплексное действие света и температуры может быть не только положительным, но и отрицательным. Сильный [солнечный свет при высокой температуре](http://vinogradna.ru/vliyanie-vysokoj-temperatury-na-vinogradnoe-rastenie.html) может привести к ожогу листьев, побегов и ягод. Особенно вредно отражается сильный солнечный свет при резкой смене освещения. Этими причинами и объясняется большой процент гибели [почек](http://vinogradna.ru/stroenie-glaza-i-pochki.html), распустившихся в земле (до открывки виноградников), которые после открывки сразу попадают под действие прямых солнечных лучей.

**Давление**

Скорость, с которой растут растения, зависит от условий атмосферного давления. Если атмосферное давление слишком низкое, растение не может выжить из-за отсутствия газообмена. Атмосферное давление важно для питания растущих растений.

**Погодные явления**

Серьёзно влияют на урожай и разнообразные погодные явления — ветер, заморозки, град, снег и прочие. Так, влажный ветер с моря способствует лучшему наливу ягод в фазе созревания, сухой континентальный — повышает сахаристость. В то же время при повышенной влажности, ветра могут переносить грибковые споры, которые способствуют распространению болезней. Легкий ветер в период активной вегетации благоприятно влияют на аэрацию листьев, перенос пыльцы.

Неблагоприятны для качества винограда заморозки, в особенности — в фазе созревания ягод. Заморозки задерживают созревание и мешают накоплению сахар. Подмороженные ягоды, меняют свой цвет на красно-бурый, становятся мягкими и приобретают неприятный, специфический привкус. Весьма опасен для урожая винограда град. Глубоко поврежденые градом недозрелые ягоды останавливают развитие, засыхают; вино из них, зачастую, приобретает неприятную терпкость, горечь и специфический привкус.

**Влажность**

Влажность в комплексе с температурой в наибольшой степени влияет на то, как проходит вегетация виноградного растения и на качество урожая.

Значение гидротермического коэффициента (ГТК) за период май—июль используют при культивировании винограда как показатель влагообеспеченности растения служит.

По изолинии ГТК 0,5 проводится условная граница между орошаемой и неорошаемой зонами виноградарства.

# 3. Откуда берем погодные данные

**Почему openweathermap**

Данные о погоде в быстром и удобном виде. Предоставляют хорошо узнаваемые погодные продукты, которые значительно упрощают работу с данными о погоде.Openweathermap круглосуточно работают с миллионами разработчиков и считают, что эти преимущества могут подойти для большинства приложений, вплоть до сложных корпоративных систем.

**Спектр готовых**

1. использовать погодные продукты
2. Краткосрочные и долгосрочные прогнозы, история и наблюдения
3. Любое место на земном шаре
4. Прозрачное ценообразование и лицензирование

Доступ к метеорологическим продуктам осуществляется через быстрые и надежные API-интерфейсы, соответствующие отраслевым стандартам и совместимые с различными корпоративными системами. Все продукты содержат важные климатические параметры для любого места. Кроме того, есть несколько специальных продуктов, полезных для отраслей, на которые влияют погодные условия, например, API дорожного риска, API солнечной радиации и т. д. Для каждой точки земного шара OpenWeather предоставляет гиперлокальный минутный прогноз, исторические данные, текущее состояние и от краткосрочных до годовых и прогнозируемых данных о погоде. Все данные доступны через стандартные отраслевые API. В своем проекте для сбора статистики использую исторические данные

**Историческая коллекция погоды**

1. Архив истории за более чем 40 лет с почасовой детализацией
2. История прогнозов Архив предыдущих прогнозов, сделанных на 16 дней вперед для любой точки земного шара
3. History API с данными о погоде за месяц или год назад в зависимости от вашей подписки.
4. Statistical Weather API со статистическими данными по основным параметрам погоды на выбранную дату или месяц года, либо за весь год

# 4. Влияние почвенных параметров

Данные берутся средние(mean)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Параметры почвы | Акроним | Единицы измерения | Сопоставленные единицы | Описание |
| Объемная плотность | BDOD | kg/dm3 | cg/cm3 | Насыпная масса мелкоземной фракции в сухом состоянии |
| Катионный обмен | CEC | cmol(c)/kg | mmol(c)/kg | Способность фракции мелкозема удерживать обменные катионы |
| Крупные фрагменты | CFVO | cm3 /100 cm3 (volume %) | cm3 /dm3 | Объемное содержание фрагментов крупнее 2 мм во всем грунте |
| Нитрогены | N | g/kg | cg/kg | Сумма общего азота (аммиак, органический и восстановленный азот), измеренная методом Кьельдаля, плюс нитрат-нитрит. |
| рН | pH | – | 10∗ | Отрицательный десятичный логарифм активности ионов гидроксония ( H+) в воде |
| Концентрация органического углерода | SOC | g/kg | dg/kg | Гравиметрическое содержание органического углерода в мелкоземной фракции почвы |
| Фракция текстуры почвы | STF | % | g/kg | Гравиметрическое содержание песка, ила и глины в мелкоземистой фракции почвы |

**Объемная плотность, Крупные фрагменты**

Механический состав почв и пород влияет на развитие корневой системы винограда, степень ее разветвления и глубину проникновения. Чем тяжелее почвы, тем длиннее скелетные корни, выше их масса и тем меньше на них обрастающих корешков. Причина в том, что тяжелые почвы имеют высокое механическое сопротивление. Чтобы преодолеть его, т. е. проникнуть сквозь частички земли вниз, виноград развивает толстые, похожие на земляной бур, корни. Однако силы винограда не бесконечны. Плотность некоторых почв достигает критических показателей, при которых нормальное развитие корневой системы винограда становится уже невозможной. Величина этого показателя зависит от типа почвы. На слишком плотных и твердых почвах виноград дает очень скромные урожаи. Сахаристость ягод при этом низкая, а кислотность, напротив, повышенная.

Другой фактор, влияющий на глубину залегания и ширину разрастания корневой системы винограда, — *аэрация почвы*, т. е. уровень содержания в почвенном воздухе кислорода, углекислого газа, а также сероводорода, водорода, метана и других побочных продуктов анаэробного разложения. Виноград в этом отношении очень требователен. При недостаточной аэрации корни хуже поглощают питательные вещества и воду, находящиеся в почве. Плохими агрофизическими свойствами обладает и слитый чернозем, содержащий более 65 % глины и 41 % ила. На таких почвах длина виноградных корней в 5 раз меньше, урожайность в 2–3 раза ниже нормы, сахаристость ягод падает на 2–4 %, а кислотность повышается на 1–2 %.

Для посадки винограда непригодны участки с непроницаемой для корней подпочвой, со скалистыми породами, залегающими ближе 1 м к поверхности, а также с заболоченной, засоленной, излишне каменистой и щебенчатой почвой.

Неблагоприятны для роста и урожайности винограда почвы с плотностью выше 1,4 г/см3, твердостью более 20 кг/см2, пористостью при влажности менее 15 %.

На сроки и скорость вегетации винограда влияет *температура почвы*, которая зависит от количества поглощенной солнечной энергии. Скорость прогревания почвы связана с ее цветом. Как известно, темный цвет притягивает больше лучей, чем светлый. Поэтому черноземные, каштановые и другие темные почвы прогреваются раньше и сильнее светлых известковых, песчаных, каменистых. Следовательно, и виноград на них развивается быстрее и созревает раньше.

Конечно, характер почв всегда учитывается при выращивании культур, будь то плодово-ягодные деревья и кустарники, злаковые, бахчевые, овощные культуры или цветы. Но при выращивании винограда учитывать характеристики почвы особенно важно. Установлено, что особенности почвы влияют на вкус виноградного сока, а следовательно, и на аромат вина, которое изготавливают из этого сока. Выращивая виноград на определенных почвах, можно получить вино с характерными качествами.

**Катионный обмен**

Катионообменную способность почв относят к числу фундаментальных свойств почвы. Катионный обмен – это обратимый процесс стехиометрического обмена ионами между двумя контактирующими фазами.

Состав обменных катионов почв является одним из важнейших показателей, используемых при диагностике и классификации почв. Так, например, в зависимости от содержания катионов Н+ и А13+все почвы можно разделить на две группы: почвы, насыщенные основаниями (не содержат Н+ и А13+), и почвы, не насыщенные основаниями (содержат Н+ и А13+).

Почвы, не насыщенные основаниями, содержат некоторые количества обменных катионов Н+ и А13+, и для них всегда S*<*ЕКО. В эту группу почв входят подзолистые, дерново-подзолистые, болотные, серые и бурые лесные почвы, некоторые черноземы и почвы влажных субтропиков.

Насыщенные основаниями почвы – это преимущественно степные почвы (черноземы, каштановые, сероземы, бурые и серо-бурые степные), а также почвы различных зон, сформированные при участии жестких грунтовых вод или на карбонатных породах. Среди этой группы следует выделитьпочвы, содержащие в значительных количествах обменный катион Na+. К ним относятся солонцеватые почвы, солонцы, многие солончаки. Повышенное содержание обменного катиона натрия в солонцах, например, приводит к появлению таких отрицательных свойств почв, как высокая щелочность, плотная слеживаемость в сухом состоянии, сильная трещиноватость почвенной корки, что вызывает разрыв корневой системы растений и гибель всходов; во влажном состоянии такие почвы отличаются плохими водно-физическими свойствами (вязкость, заплывание и липкость почвенной массы).

**рН**

Кислотность почвы имеет основополагающее значение для успеха всего предприятия по выращиванию винограда. Ее показатель – pH, может иметь нейтральное, кислое или щелочное значение.

Традиционно считается, что самый благоприятный для выращивания растений уровень кислотности колеблется в интервале между показателям 4.0 и 8.0. Более высокая степень кислотности негативно отражается на способности винограда усваивать питание, получаемое из почвы.

**Концентрация органического углерода**

Чем больше разница между долей азота и углерода, тем лучше для винограда. Например, предкавказские черноземы содержат достаточно много азота, поэтому из местного винограда обычно получаются не самые лучшие вина, однако в отдельные годы, в которые наблюдаются умеренные осадки, ясная и продолжительная осень, качество вина, особенно красного, увеличивается в разы. Из этого следует вывод, что даже на почвах с повышенным содержанием азота и углерода можно добиться обильных и качественных урожаев винограда при условии достаточной солнечной освещенности и влажности.

**Фракция текстуры почвы**

**Лучше всего виноград растет на легких почвах, которые дают корням кислород и воду.** Оптимальные варианты – суглинки, глина, черноземы, песок и супесчаные смеси.

Лучший грунт для выращивания винограда – смешанный из камней, глины, песка, минеральных веществ и органики. Тяжелые глинистые почвы практически не проводят воздух и воду, поэтому для закладки виноградника не подходят. Песчаные быстро отдают влагу, промерзают в холода, что приводит к гибели молодых растений. Также грунт должен содержать железо, азот, кальций, калий, магний, фосфор – дефицит данных элементов негативно влияет на рост растений.

* **азот** ― влияет на рост, избыток элемента ведет к увеличению зеленой массы и снижению урожайности, недостаток ― замедляет развитие;
* **железо** ― стимулирует выброс хлорофилла;
* **калий**― отвечает за накопление сахара и крахмала, увеличивает морозостойкость, ускоряет обмен веществ, его нехватка чревата образованием кислых ягод;
* **кальций** ― необходим для развития корневой системы, но избыток приводит к хлорозу;
* **магний**― участвует в образовании хлорофилла, при его недостатке листья желтеют и опадают;
* **фосфор**― отвечает за плодоношение;
* **сера**― важный питательный элемент.

# 5. Откуда берутся почвенные данные

**Информация SoilGrid**

Здоровые почвы обеспечивают важные экосистемные услуги на местном, ландшафтном и глобальном уровнях и важны для функционирования наземных экосистем. Информация о мировых почвенных ресурсах, основанная на «наилучших доступных» (общих) данных профиля почвы в масштабе, соответствующем потребностям пользователей, необходима для решения ряда насущных глобальных проблем. К ним относятся предотвращение и сокращение эрозии почвы за счет восстановления и освоения земель, смягчение последствий изменения климата и адаптация к ним и обеспечение водной безопасности производства продуктов питания и продовольственной безопасности

**Почему SoilGrid**

До последнего десятилетия в большинстве оценок глобального масштаба, требующих данных о почве, использовалась Цифровая карта почв мира (DSMW), обновленная версия оригинала, напечатанная в масштабе 1:5 × 106 масштабная карта почв мира (SMW). Почвенно-географические данные из DSMW послужили основой для создания полученных данных о свойствах почвы, основанные на более широком наборе данных о профилях почв, хранящихся в базе данных WISE, и более сложных процедурах (таксотрансфер) для получения различных свойств почвы. Впоследствии в результате совместных усилий, координируемых Продовольственной и сельскохозяйственной организацией Объединенных Наций (ФАО), наилучшая доступная (более новая) информация о почвах, собранная для центральной и южной части Африки, Китая, Европы, северной Евразии и Латинской Америки, была объединена в новый продукт, известный как Гармонизированная мировая база данных о почвах (HWSD). До недавнего времени HWSD была единственной базой данных приложений к цифровым картам, доступной для глобального анализа. Однако он имеет ряд ограничений. Некоторые из них связаны с частично устаревшими почвенно-географическими данными, а также с использованием двухслойной модели (0–30 и 30–100 см) для получения свойств почвы. Другие касаются самих производных атрибутивных данных, в частности их неопределяемой количественно неопределенности и использования трех различных версий легенды ФАО (т.е. ФАО74, ФАО85 и ФАО90). Эти вопросы в той или иной степени были решены в различных новых глобальных наборах почвенных данных (Batjes, 2016; Shangguan et al., 2014; Stoorvogel et al., 2017), которые по-прежнему в значительной степени основаны на традиционном подходе к картографированию почв. В последнее десятилетие цифровое картографирование почв (DSM) стало широко использоваться для получения информации о почвах. DSM состоит в первую очередь в построении количественной численной модели на основе наблюдений за почвой и информации об окружающей среде, выступающей в качестве заменителей факторов почвообразования. DSM также может интегрировать прямую информацию в качестве заменителей свойств почвы, например, измерения проксимального зондирования. Количество исследований с использованием DSM для создания карт свойств почвы постоянно растет. Рассматриваются многочисленные подходы к моделированию, от линейных моделей до геостатистики, машинного обучения и искусственного интеллекта (например, глубокого обучения). SoilGrid это, набор новых карт свойств почв для мира с разрешением сетки 250 м с процессом, включающим самые современные методы и адаптирующим их к задачам глобального цифрового картографирования почв с использованием устаревших данных. Он основан на предыдущих глобальных картах свойств почвы (SoilGrids250m) (Hengl et al., 2017b), интегрируя современные методы машинного обучения, высокая доступность стандартизированных данных о профиле почвы по всему миру и экологические ковариаты. В частности, рассматриваются следующие элементы в глобальном масштабе:

1. включение данных о профилях почв, полученных из Всемирной службы информации о почвах (WoSIS) IS-RIC, с расширенным количеством и пространственным распределением наблюдений.

2. воспроизводимая процедура выбора ковариат, основанная на рекурсивном исключении признаков.

3. улучшенная процедура перекрестной проверки, основанная на пространственной стратификации.

4. Количественная оценка неопределенности прогноза с использованием квантильных регрессионных лесов.

**Свойства почвы**

Для целей SoilGrids «почва» — это рыхлый материал толщиной до 2 м в эпидермисе Земли, находящийся в прямом контакте с атмосферой; таким образом, подводные почвы и почвы, подверженные воздействию приливов и отливов, здесь не рассматриваются. Ни материалы глубже 2 м. Что для на крайне удобно так как в большинстве случаях корни винограда глубже не растут. Это решение имеет последствия для расчетов общих запасов, в частности органического углерода почвы. В таблице 1 описаны свойства почвы, учитываемые в этой версии SoilGrids: содержание органического углерода, общее содержание азота, рН почвы (измеренный в воде), способность к обмену катионов, гранулометрический состав почвы и доля крупных фрагментов. «Подстилочные слои» поверх минеральных почв были исключены из дальнейшего моделирования с использованием следующих допущений. Согласованность глубины слоя (например, последовательное увеличение верхней и нижней глубины для каждого слоя вниз по профилю) в WoSIS проверялась с использованием автоматизированных процедур. В соответствии с действующими международно-признанными соглашениями такие приращения глубины даются как «измеренные от поверхности, включая органические слои и минеральные покровы» (ФАО, 2006; Schoeneberger et al., 2012).

Тем самым если не были получены данные почвы, то считается что в этом месте нельзя выращивать виноград (вода, приливы отливы, города, горы и т.д.)

**Предсказание свойств почвы**

Был рассчитан процент перекрестных проверок, содержащихся в интервале прогнозирования 0,9 В идеале PICP близок к 0,9, что указывает на правильную оценку неопределенности. Значение PICP, существенно превышающее 0,9, свидетельствует о том, что неопределенность была недооценена; существенно меньший PICP указывает на то, что он был завышен. 90-й интервал предсказания (PI90)

PI90 =q0.95−q0.05

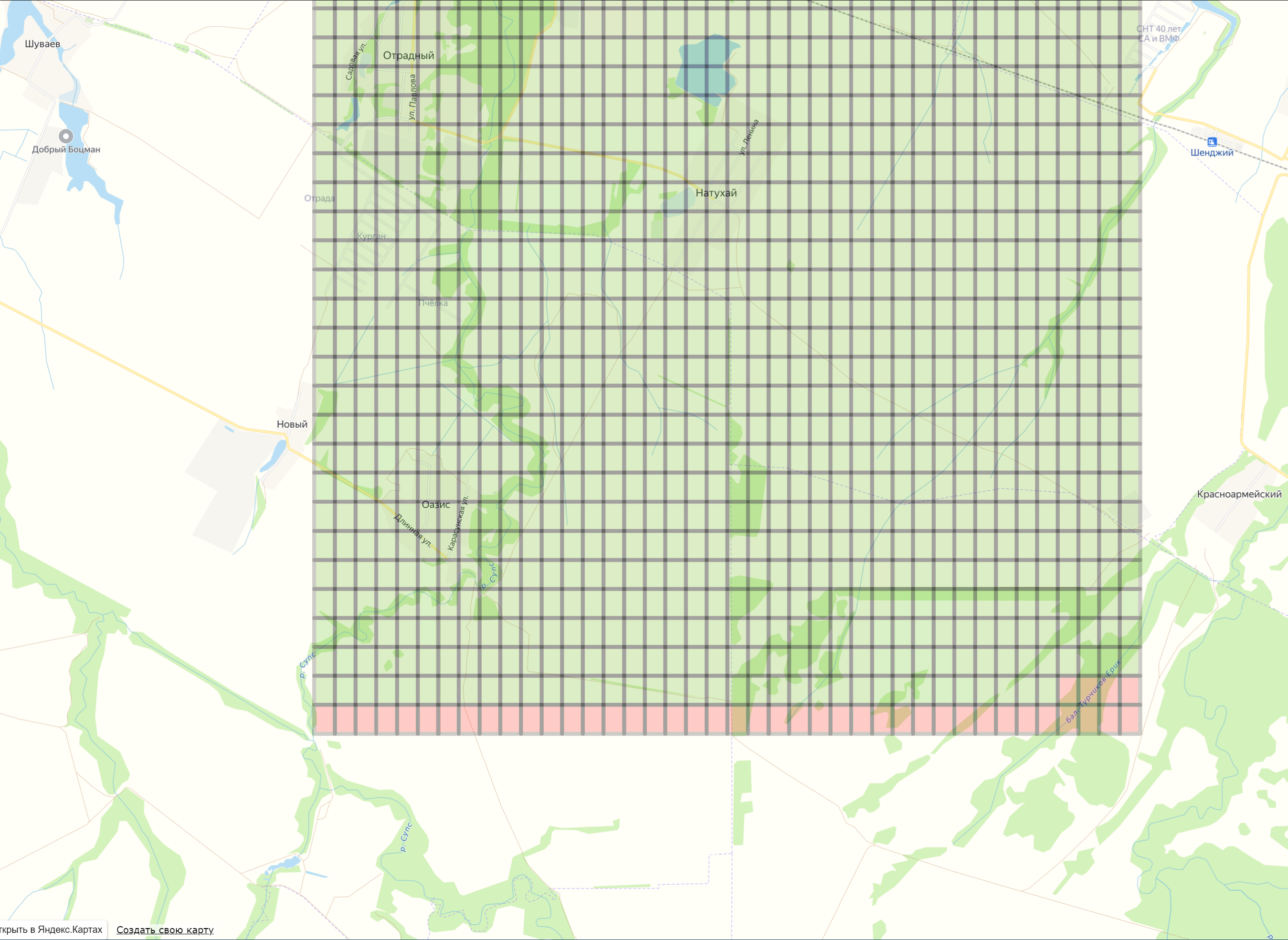
# 6. Функционал приложения

# 6.1 Фронтенд

Интерфейс реализован через javascript с использованием Фреймворка React.

Интерфейс представляет из себя карту с набором полигонов\* различного цвета в зависимости от скоринга\*. Фронт состоит из двух частей:

**Компонент карты**

****

Компонент карты – это фреймворк Ymap который размещает интерактивную карту на веб-страницу. При иницилализации карты указывается контейнер в котором она размещается и передаем туда опции:

1. Начальное положение, центральное положение карты которое будет отображаться, устанавливаем значение координат центрального полигона
2. Размер карты
3. Действие при нажатие на карту. В моем случае при нажатие на карту убирается отображение бокового меню

Карта содержат произвольное количество слоев равное количеству полигонов. Слой — это визуальный компонент, который отвечает за отрисовку слоя определенного содержания. В моем случае слой представляет из себя компонент Полигон, с настройками:

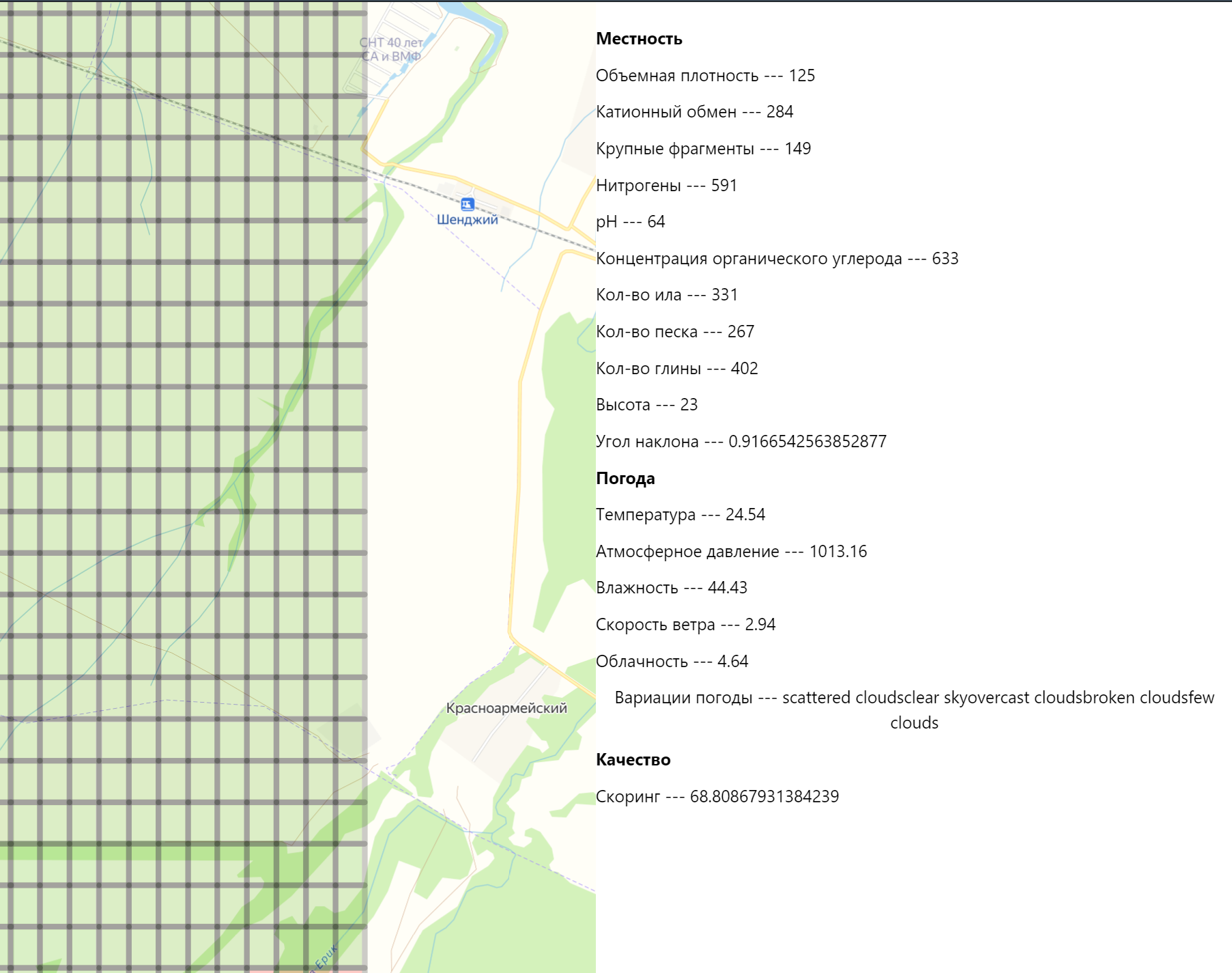
1. Цвет полигона, для визуального определения его пригодности
2. Цвет границы полигона, тип границы, толщина границы. Чтобы видеть где находятся его границы
3. Прозрачность полигона, чтобы увидеть какие объекты находятся в полигоне, например: реки, дома и т.д.
4. Действие при нажатие на полигон. В моем случае отображается боковое меню с его характеристиками
5. Расположение полигона. Значение координат его угловых точек

При загрузки главной страницы фронт делает запрос на бек для получение всех полигонов и запись эти данных на стороне фронта

**Компонент полигон**

Компонент полигон содержит данные по выбранному полигону которые отображается в боковом меню:

1. Высота
2. Максимальный угол наклона
3. Температура
4. Атмосферное давление
5. Влажность
6. Скорость ветра
7. Облачность
8. Вариации погоды
9. Кол-во песка в почве
10. Кол-во глины в почве
11. Значение ph
12. Концентрация органического углерода
13. Значение нитрогенов
14. Объемная плотность
15. Катионный обмен
16. Кол-во крупных фрагментов
17. Кол-во ила в почве
18. Значение скоринга



# 6.2 Бэкенд

Бэкенд приложение написан на python с использованием фреймворка DRF. Взамодействие с бэкендом просиходит через API которые предоставляет бэкенд, по http протоколу

**API**

‘/polygons’ GET – Получение всех полигонов

Response 200: list(

Id: int, номер полигона

lat: float, широта центральной точки полигона

lng: float, долгота центральной точки полигона

x1: float, широта верхней левой точки полигона

y1: float, долгота верхней левой точки полигона

x2: float, широта верхней правой точки полигона

y2: float, долгота верхней правой точки полигона

x3: float, широта нижний правой точки полигона

y3: float, долгота нижний правой точки полигона

x4: float, широта нижний левой точки полигона

y4: float, долгота нижний левой точки полигона

elevation: int, высота

inclination: float, угол наклона

temp: float, температура

pressure: float, давление

humidity: float, влажность

wind\_speed: float, скорость ветра

wind\_gust: float, ускорение ветра

clouds: float, облачность

weather: list, Вариации погоды

bdod: int, Объемная плотность

cec: int, Катионный обмен

cfvo: int, Крупные фрагменты

clay: int, Кол-во глины

nitrogen: int, Нитрогены

ocd: int, Плотность органического углерода

ocs: int, Запас органического углерода

phh2o: int, рН

sand: int, Кол-во песка

silt: int, Кол-во ила

soc: int, Концентрация органического углерода

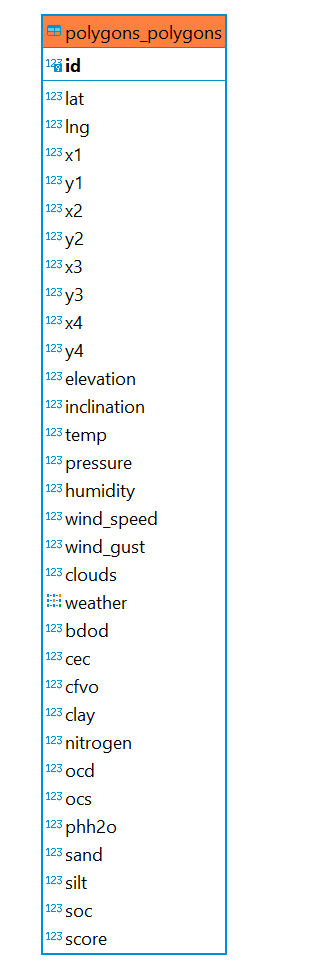
score: float, Скоринг

)

**БД**

База данных реализована на PostgreSQL и содержит таблицы:

1. Заполнение таблицы происходит путем миграции – читается файл с данными местности и записываются в бд



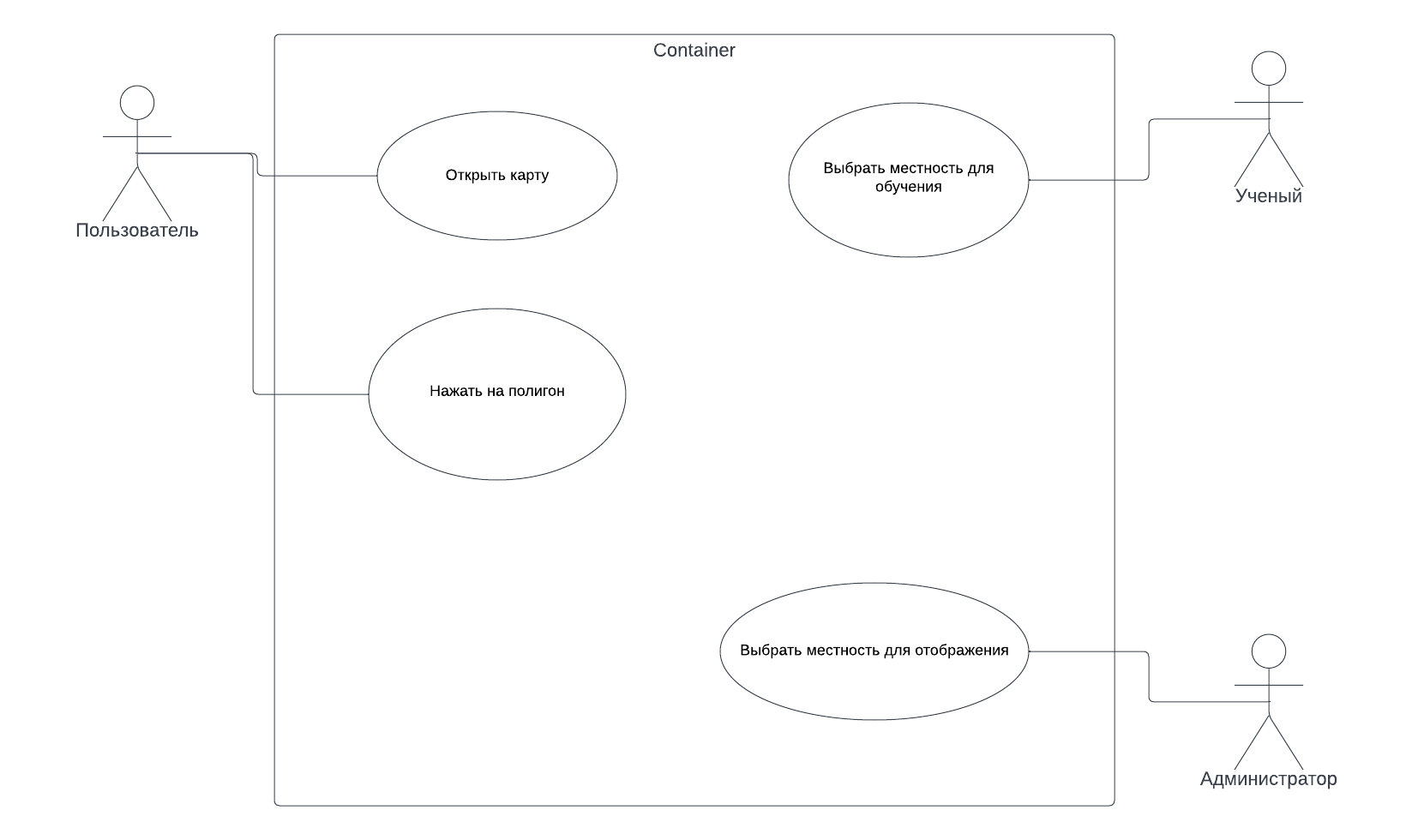
**Миграция в бд**

Миграция в бд происходит через команду python manage.py makemigrations которая создает таблицы в бд по описанию и через команду python manage.py migrate которая запускает скрипт для заполнение таблиц данными. Я написал скрипт который читает файл с наборами полигонов и переносит эти данные в таблицу

**Взаимодействие с приложением**

Взаимодействие с приложение я отобразил на схеме UseCase. У меня реализованы различные роли которые по разному могут взаимодействовать с приложением.

1. Пользователь: Посмотреть на карте полигоны и нажать на любой полигон чтобы посмотреть на его характеристики
2. Ученый – Выбор местность путем указание координат начальной точки, размера местности, указания признака пригодности для выращивания , для дальнейшего сбора климатических и почвенных данных. Эти полигоны передаются для машинного обучение модели
3. Администратор - Выбор местность путем указание координат начальной точки, размера местности. Эта местность будет отображаться на карте



# 6.2.1 Взаимодействие администратора с приложением

Взаимодействие с приложением у администратор реализовано через контроллер AdminController. При инициализации контроллера задается путь сохранения местности в виде файла для ее дальнейшей миграции в базу данных. Для формирования местности в контреллеры вызываются последовательно методы:

1. Метод create\_polygons служит для создание полигонов и записью в него начальных данных. В нем инициализируется сущность CreatePolygons, в нее передаются стартовые координаты, размер местности, размер полигона. В этой сущности вызывается метод create\_polygons который генерирует набор полигонов по переданным параметрам и записывает этот набор в словарь:

{

(lat: float, lng: float):

{

X1,

Y1,

X2,

Y2,

X3,

Y3,

X4,

Y4,

}, …

}

Далее в этой же сущности вызывается метод set\_elevation куда передается набор полигонов. В этом методе происходит итерация по наборам полигонов и для каждого полигона по API c <https://api.airmap.com/elevation/v1/ele/> берут значение высоты и сохраняют его для полигона как {elevation: int}.

Далее в этой же сущности вызывается метод set\_elevation куда передается набор полигонов. В этом методе происходит итерация по наборам полигонов и для каждого полигона выбираются значение высоту у соседних полигонов расположенных горизонтально и вертикально, и значение высоты текущего полигона. По этим параметрам определяется угол наклона по формуле(1)

Где b – значение расстояние между центральными точками полигонов, height – разница высоты между текущим полигоном и соседним полигоном.

Угол наклона вычисляется у каждого соседа и выбирается максимальное значение. Это значение записывается в виде {inclination: float}

1. Метод get\_polygons служит для наполнение полигона почвенными, климатическими характеристиками и получение скоринга путем итерации по наборам полигонов и взаимодействие с полигоном находящимся в цикле. Получение этих данных происходит в сущности PolygonAdminEnt куда при инициализации передается полигон. В этой сущности вызываются методы:
2. метод set\_weather служит для получение данных по API с <https://history.openweathermap.org/data/2.5/history>, с диапазон времени неделя, с шагом каждый час, с выбранным месяцем июнь, по среднему результату. Данные которые мы получаем: Температура, давление, влажность, скорость ветра, порыв ветра, облачность %, вариация погоды. Эти параметры записываются в полигон
3. метод set\_soil служит для получение данных по API с <https://rest.isric.org/soilgrids/v2.0/properties/query>, глубина почвы выбирается от 0 – 5 метров, точность выбирается 95%, берутся усреднённые данные. Данные которые мы получаем: Объемная плотность, Катионный обмен, Крупные фрагменты, Кол-во глины, Нитрогены, Плотность органического углерода, Запас органического углерода, рН, Кол-во песка, Кол-во ила, Концентрация органического углерода. Эти параметры записываются в полигон
4. метод scored\_polygons служит для получение скоринга от модели. Для начало преобразуем данные в модели-читаемый формат. Убираем данные которые не участвуют в получение скоринга. Получаем от модель скоринг. Объеденяем данные которые участвыволи в получение скоринга, скоринг и данные которые не участвовали в получение скоринга. Изменяем оценку по фичам: если угол наклона более 45 градусов на это полигоне нельзя выращивать виноград и выставляем ему значение скоринга равным = 0. Параметр скоринга записывается в полигон.
5. Метод save\_polygons служит для сохранения набора полигонов в файл для дальнейшй миграции в базу данных

# 6.2.2 Взаимодействие ученого с приложением

Взаимодействие с приложением у ученого реализовано через контроллер ScinceController. При инициализации контроллера задается путь сохранения набора полигонов и признак выращивания. Для формирования местности в контроллеры вызываются последовательно методы:

1. Метод create\_polygons служит для создание полигонов и записью в него начальных данных. В нем инициализируется сущность CreatePolygons, в нее передаются стартовые координаты, размер местности, размер полигона. В этой сущности вызывается метод create\_polygons который генерирует набор полигонов по переданным параметрам и записывает этот набор в словарь:

{

(lat: float, lng: float):

{

X1,

Y1,

X2,

Y2,

X3,

Y3,

X4,

Y4,

}, …

}

Далее в этой же сущности вызывается метод set\_elevation куда передается набор полигонов. В этом методе происходит итерация по наборам полигонов и для каждого полигона по API c <https://api.airmap.com/elevation/v1/ele/> берут значение высоты и сохраняют его для полигона как {elevation: int}.

Далее в этой же сущности вызывается метод set\_elevation куда передается набор полигонов. В этом методе происходит итерация по наборам полигонов и для каждого полигона выбираются значение высоту у соседних полигонов расположенных горизонтально и вертикально, и значение высоты текущего полигона. По этим параметрам определяется угол наклона по формуле(1)

Где b – значение расстояние между центральными точками полигонов, height – разница высоты между текущим полигоном и соседним полигоном.

Угол наклона вычисляется у каждого соседа и выбирается максимальное значение. Это значение записывается в виде {inclination: float}

1. Метод get\_polygons служит для наполнение полигона почвенными, климатическими характеристиками и признаком выращиваемости путем итерации по наборам полигонов и взаимодействие с полигоном находящимся в цикле. Получение этих данных происходит в сущности PolygonScinceEnt куда при инициализации передается полигон и признак выращиваемости. В этой сущности вызываются методы:
2. метод set\_weather служит для получение данных по API с <https://history.openweathermap.org/data/2.5/history>, с диапазон времени неделя, с шагом каждый час, с выбранным месяцем июнь, по среднему результату. Данные которые мы получаем: Температура, давление, влажность, скорость ветра, порыв ветра, облачность %, вариация погоды. Эти параметры записываются в полигон
3. метод set\_soil служит для получение данных по API с <https://rest.isric.org/soilgrids/v2.0/properties/query>, глубина почвы выбирается от 0 – 5 метров, точность выбирается 95%, берутся усреднённые данные. Данные которые мы получаем: Объемная плотность, Катионный обмен, Крупные фрагменты, Кол-во глины, Нитрогены, Плотность органического углерода, Запас органического углерода, рН, Кол-во песка, Кол-во ила, Концентрация органического углерода. Эти параметры записываются в полигон
4. метод train\_polygons служит для выставление признака выращивания для полигона
5. Метод save\_polygons служит для сохранения набора полигонов в файл для дальнейшй миграции в машинное обучение

# 7. Машинное обучение

Для машинного обучения я использую ансамблевые методы, а именно модель дерева решений random forest classifier.

**Ансамблевые методы**

Цель **ансамблевых методов** — объединить прогнозы нескольких базовых оценок, построенных с заданным алгоритмом обучения, чтобы улучшить обобщаемость / надежность по сравнению с одной оценкой.

В ансамблевых алгоритмах методы упаковки образуют класс алгоритмов, которые строят несколько экземпляров оценщика черного ящика на случайных подмножествах исходного обучающего набора, а затем агрегируют свои индивидуальные прогнозы для формирования окончательного прогноза. Эти методы используются как способ уменьшить дисперсию базовой оценки (например, дерева решений) путем введения рандомизации в процедуру построения и последующего создания из нее ансамбля. Во многих случаях методы упаковки представляют собой очень простой способ улучшения по сравнению с одной моделью, без необходимости адаптации базового алгоритма. Поскольку они обеспечивают способ уменьшения переобучения, методы упаковки лучше всего работают с сильными и сложными моделями (например, полностью разработанными деревьями решений), в отличие от методов повышения, которые обычно лучше всего работают со слабыми моделями (например,неглубокие деревья решений).

Методы упаковки бывают разных видов, но в основном отличаются друг от друга тем, как они рисуют случайные подмножества обучающего набора:

* Когда случайные подмножества набора данных рисуются как случайные подмножества выборок, этот алгоритм известен как Вставка [[B1999]](https://scikit-learn.ru/1-11-ensemble-methods/#link) .
* Когда образцы отбираются с заменой, этот метод известен как Bagging [[B1996]](https://scikit-learn.ru/1-11-ensemble-methods/#link) .
* Когда случайные подмножества набора данных рисуются как случайные подмножества признаков, этот метод известен как случайные подпространства [[H1998]](https://scikit-learn.ru/1-11-ensemble-methods/#link) .
* Наконец, когда базовые оценки строятся на подмножествах как выборок, так и функций, тогда метод известен как случайные исправления [[LG2012]](https://scikit-learn.ru/1-11-ensemble-methods/#link) .

В scikit-learn методы пакетирования предлагаются как унифицированная [BaggingClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html" \l "sklearn.ensemble.BaggingClassifier" \t "_blank) метаоценка (соответственно **[BaggingRegressor](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingRegressor.html" \l "sklearn.ensemble.BaggingRegressor" \t "_blank)**), принимающая в качестве входных данных определяемую пользователем базовую оценку вместе с параметрами, определяющими стратегию рисования случайных подмножеств. В частности, max\_samples и max\_features контролировать размер подмножеств (с точки зрения образцов и функций), bootstrap а также bootstrap\_features контролировать, будут ли образцы и элементы отображаться с заменой или без нее. При использовании подмножества доступных образцов точность обобщения можно оценить с помощью нестандартных образцов путем настройки oob\_score=True. В качестве примера приведенный ниже фрагмент иллюстрирует, как создать экземпляр ансамбля KNeighborsClassifier базовых оценок, каждая из которых построена на случайных подмножествах из 50% выборок и 50% функций.

Обычно выделяют два семейства ансамблевых методов:

* В **методах усреднения** главный принцип состоит в том, чтобы построить несколько оценщиков независимо, а затем усреднить их прогнозы. В среднем, комбинированная оценка обычно лучше, чем любая из оценок с одной базой, потому что ее дисперсия уменьшается. **Примеры:**[методы упаковки](https://scikit-learn.ru/1-11-ensemble-methods/#bagging) , [леса из рандомизированных деревьев](https://scikit-learn.ru/1-11-ensemble-methods/#forest) ,…
* Напротив, в **методах повышения** базовые оценки строятся последовательно, и каждый пытается уменьшить смещение комбинированной оценки. Мотивация состоит в том, чтобы объединить несколько слабых моделей для создания мощного ансамбля.

**Леса рандомизированных деревьев**

Random forest classifier - это метаоценка, которая соответствует ряду классификаторов дерева решений для различных подвыборок набора данных и использует усреднение для повышения точности прогнозирования и контроля переобучения.

Модуль **[sklearn.ensemble](https://scikit-learn.org/stable/modules/classes.html" \l "module-sklearn.ensemble" \t "_blank)** включает в себя алгоритм усреднения , основанные на рандомизированных [деревьев решений](https://scikit-learn.org/stable/modules/tree.html#tree). Алгоритм представляет собой методы «возмущать и комбинировать» [,](https://scikit-learn.ru/1-11-ensemble-methods/#link1) специально разработанные для деревьев. Это означает, что разнообразный набор классификаторов создается путем введения случайности в конструкцию классификатора. Прогноз для ансамбля дается как усредненный прогноз отдельных классификаторов.

В случайных лесах (см. [RandomForestClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html" \l "sklearn.ensemble.RandomForestClassifier" \t "_blank) и [RandomForestRegressor](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html" \l "sklearn.ensemble.RandomForestRegressor" \t "_blank) классы) каждое дерево в ансамбле строится из выборки, взятой с заменой (то есть выборкой начальной загрузки) из обучающего набора.

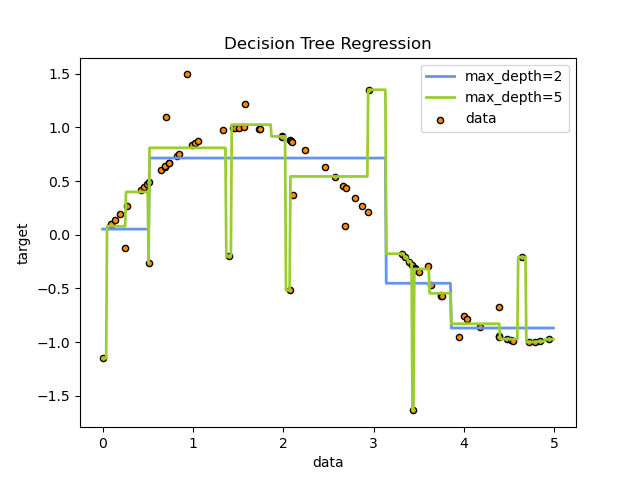
Кроме того, при разбиении каждого узла во время построения дерева наилучшее разбиение находится либо по всем входным характеристикам, либо по случайному подмножеству размера max\_features. (Подробнее см. В руководстве по [настройке параметров](https://scikit-learn.org/stable/modules/ensemble.html#random-forest-parameters) ).

Назначение этих двух источников случайности — уменьшить дисперсию оценки леса. В самом деле, отдельные деревья решений обычно демонстрируют высокую дисперсию и имеют тенденцию переоснащаться. Внедренная случайность в лесах дает деревья решений с несколько несвязанными ошибками прогнозирования. Если взять среднее значение этих прогнозов, некоторые ошибки могут быть устранены. Случайные леса уменьшают дисперсию за счет комбинирования разных деревьев, иногда за счет небольшого увеличения смещения. На практике уменьшение дисперсии часто бывает значительным, что дает в целом лучшую модель.

Мои леса строятся из деревьев DecisionTreeClassifier

**Деревья решений (DT)** — это непараметрический контролируемый метод обучения, используемый для [классификации](https://scikit-learn.ru/1-10-decision-trees/#tree-classification) и [регрессии](https://scikit-learn.ru/1-10-decision-trees/#tree-regression) . Цель состоит в том, чтобы создать модель, которая предсказывает значение целевой переменной, изучая простые правила принятия решений, выведенные из характеристик данных. Дерево можно рассматривать как кусочно-постоянное приближение.

Например, в приведенном ниже примере деревья решений обучаются на основе данных, чтобы аппроксимировать синусоидальную кривую с набором правил принятия решений «если-то-еще». Чем глубже дерево, тем сложнее правила принятия решений и тем лучше модель.



Некоторые преимущества деревьев решений:

* Просто понять и интерпретировать. Деревья можно визуализировать.
* Требуется небольшая подготовка данных. Другие методы часто требуют нормализации данных, создания фиктивных переменных и удаления пустых значений. Однако обратите внимание, что этот модуль не поддерживает отсутствующие значения.
* Стоимость использования дерева (т. Е. Прогнозирования данных) является логарифмической по количеству точек данных, используемых для обучения дерева.
* Может обрабатывать как числовые, так и категориальные данные. Однако реализация scikit-learn пока не поддерживает категориальные переменные. Другие методы обычно специализируются на анализе наборов данных, содержащих только один тип переменных. См. [Алгоритмы](https://scikit-learn.ru/1-10-decision-trees/#tree-algorithms) для получения дополнительной информации.
* Способен обрабатывать проблемы с несколькими выходами.
* Использует модель белого ящика. Если данная ситуация наблюдаема в модели, объяснение условия легко объяснить с помощью булевой логики. Напротив, в модели черного ящика (например, в искусственной нейронной сети) результаты могут быть труднее интерпретировать.
* Возможна проверка модели с помощью статистических тестов. Это позволяет учитывать надежность модели.
* Работает хорошо, даже если его предположения несколько нарушаются истинной моделью, на основе которой были сгенерированы данные.

К недостаткам деревьев решений можно отнести:

* Обучающиеся дереву решений могут создавать слишком сложные деревья, которые плохо обобщают данные. Это называется переобучением. Чтобы избежать этой проблемы, необходимы такие механизмы, как обрезка, установка минимального количества выборок, необходимых для конечного узла, или установка максимальной глубины дерева.
* Деревья решений могут быть нестабильными, поскольку небольшие изменения в данных могут привести к созданию совершенно другого дерева. Эта проблема смягчается за счет использования деревьев решений в ансамбле.
* Как видно из рисунка выше, предсказания деревьев решений не являются ни гладкими, ни непрерывными, а являются кусочно-постоянными приближениями. Следовательно, они не годятся для экстраполяции.
* Известно, что проблема обучения оптимальному дереву решений является NP-полной с точки зрения нескольких аспектов оптимальности и даже для простых концепций. Следовательно, практические алгоритмы обучения дереву решений основаны на эвристических алгоритмах, таких как жадный алгоритм, в котором локально оптимальные решения принимаются в каждом узле. Такие алгоритмы не могут гарантировать возврат глобального оптимального дерева решений. Это можно смягчить путем обучения нескольких деревьев в учащемся ансамбля, где функции и образцы выбираются случайным образом с заменой.
* Существуют концепции, которые трудно изучить, поскольку деревья решений не выражают их легко, например проблемы XOR, четности или мультиплексора.
* Ученики дерева решений создают предвзятые деревья, если некоторые классы доминируют. Поэтому рекомендуется сбалансировать набор данных перед подгонкой к дереву решений.

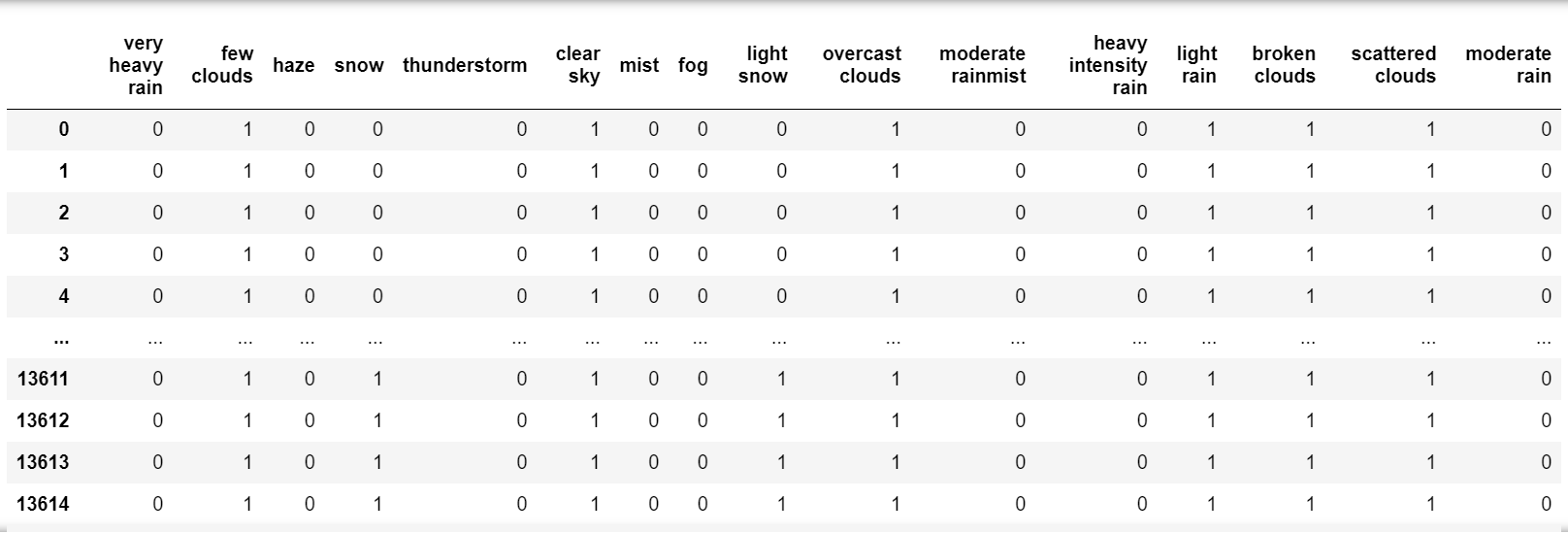
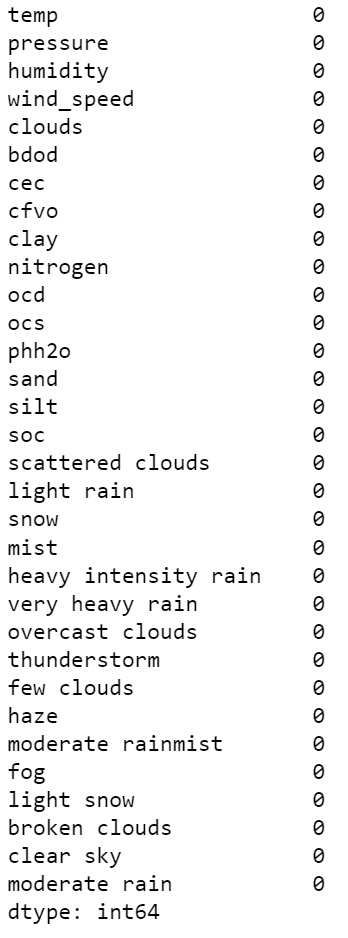
В случайном лесе я использую [**DecisionTreeClassifier**](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html#sklearn.tree.DecisionTreeClassifier) - это класс, способный выполнять мультиклассовую классификацию набора данных.

Как и в случае с другими классификаторами, [DecisionTreeClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html" \l "sklearn.tree.DecisionTreeClassifier" \t "_blank) принимает в качестве входных данных два массива: массив **X**, разреженный или плотный, формы (n\_samples, n\_features), содержащий обучающие образцы, и массив **Y** целочисленных значений, формы (n\_samples,), содержащий метки классов для обучающих образцов:

# 7.1 Преобразование данных для обучение

Для обучения необходимо загрузить данные и преобразовать их в модели-читаемый формат. С загрузкой все просто, я просто беру все датасеты и объеденяю их в один большой датасет. Для обучение участвуют различные участки с разных мест планеты, и участвуют 2 класса где растет виноград и где не может расти.

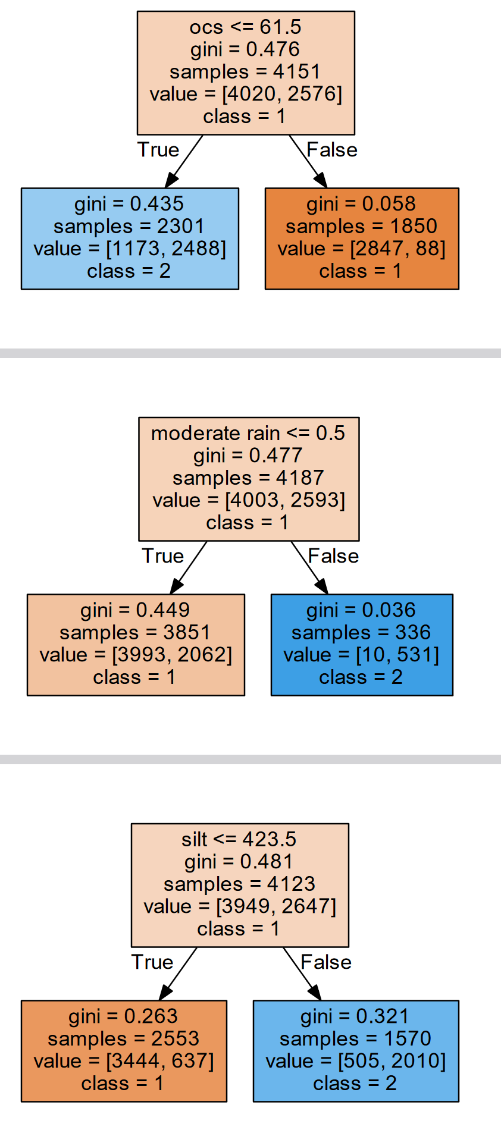
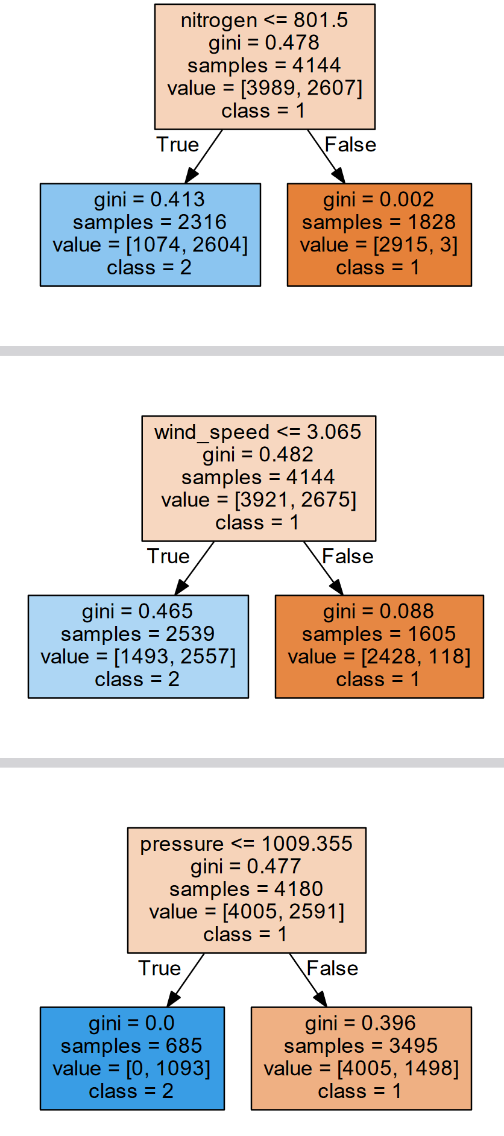
Преобразование делятся на несколько этапов:

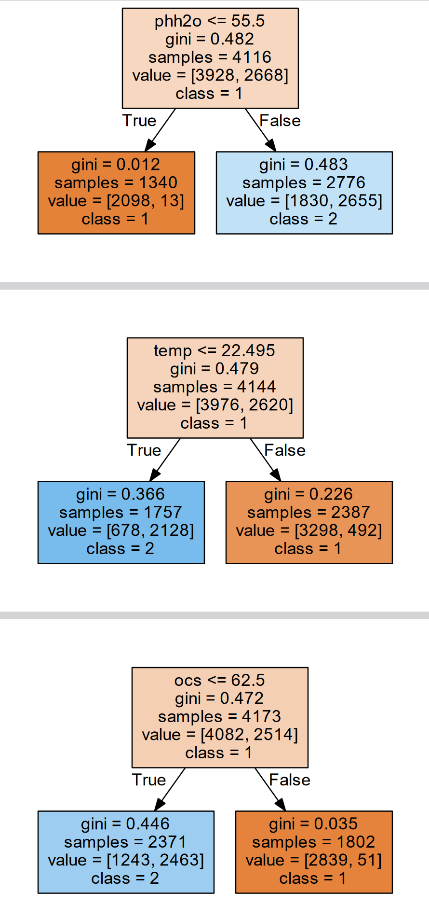
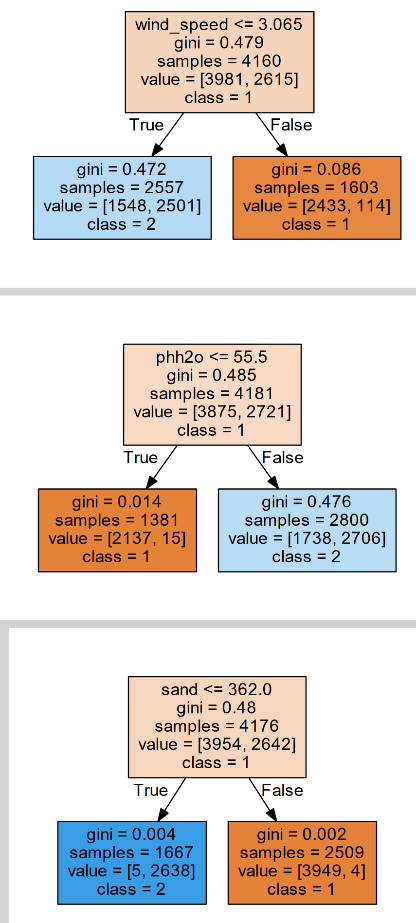
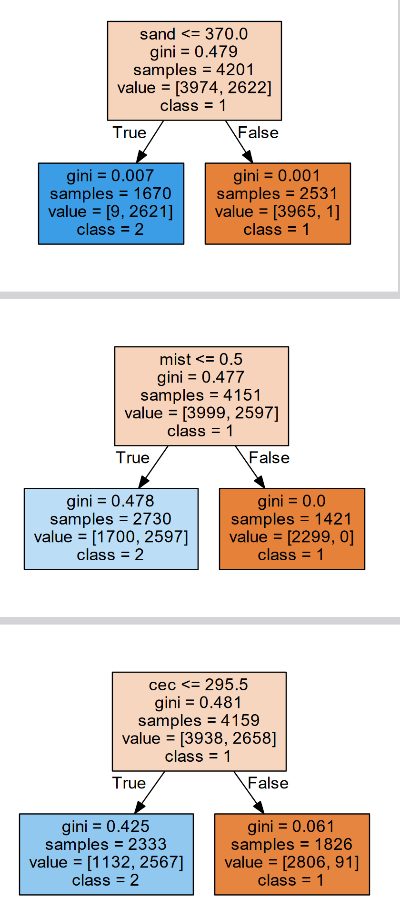
1. Преобразуем признак выращивание(growing) из bool в int (1, 0)
2. Погоду в виде списка преобразуем в колонки, которые добавляются к датасету. Если в списке была указана погода из общего списка погод, то значение в колонке будет равно 1 иначе 0 
3. Убираем из датасета колонки которые не должны участвовать в обучение: ['x1', 'y1', 'x2', 'y2', 'x3', 'y3', 'x4', 'y4', 'inclination', 'weather', 'elevation', 'wind\_gust', 'growing',]
4. Разбваем датасет на 2 части X и y. X содержат данные для обучения, y содержат класс которому относится его данные из X
5. Разбивае оба датасеты на датасеты для обучения и датасеты для проверки. X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.33, random\_state = 42, stratify=y)
6. Прверяем что в датасете нет Null значения 

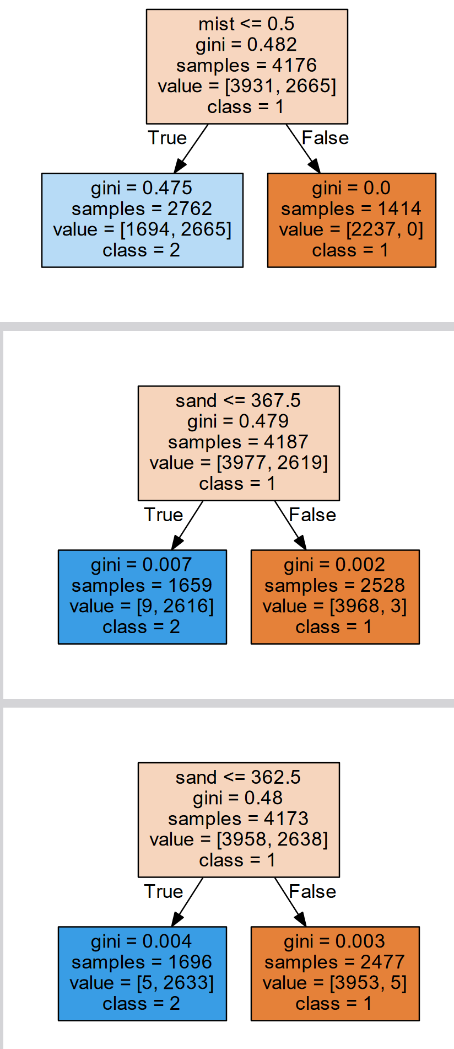
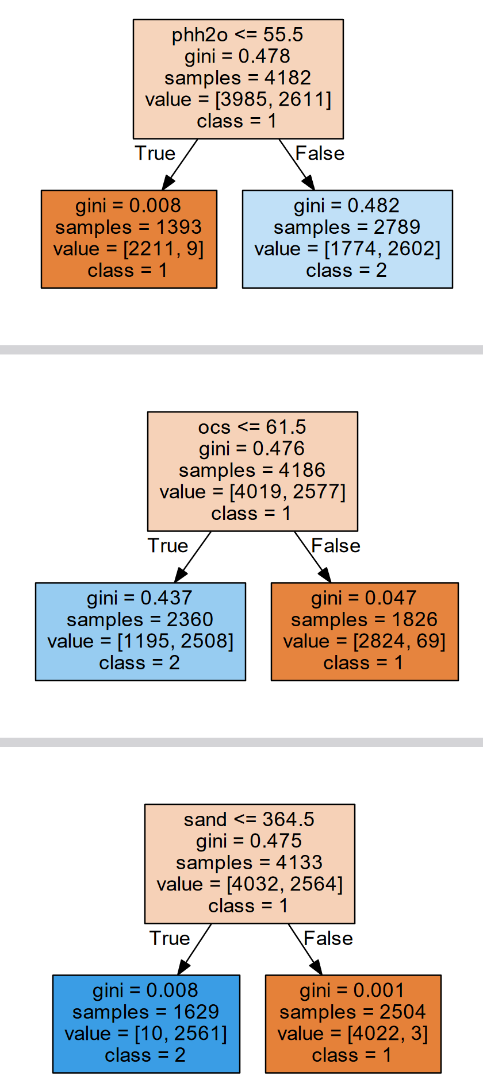
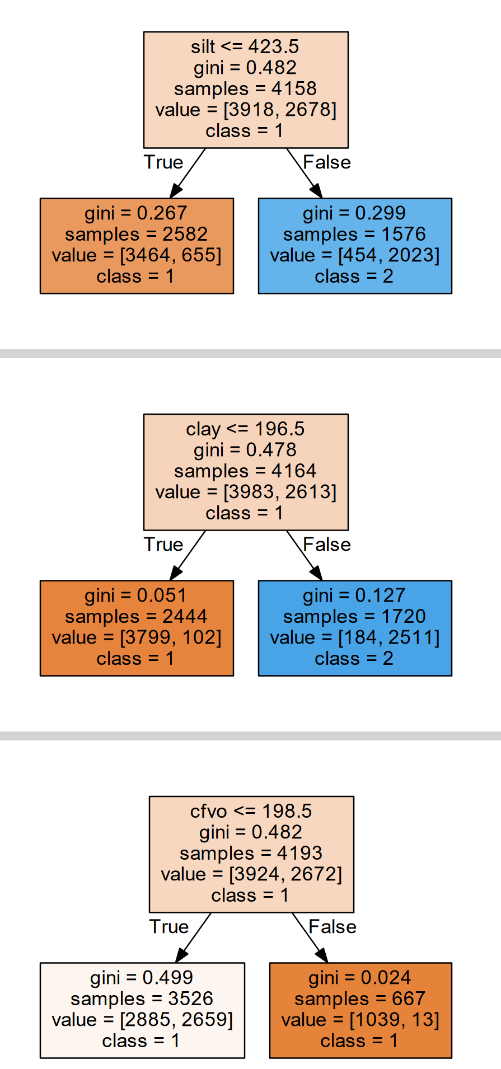
# 7.2 Обучение

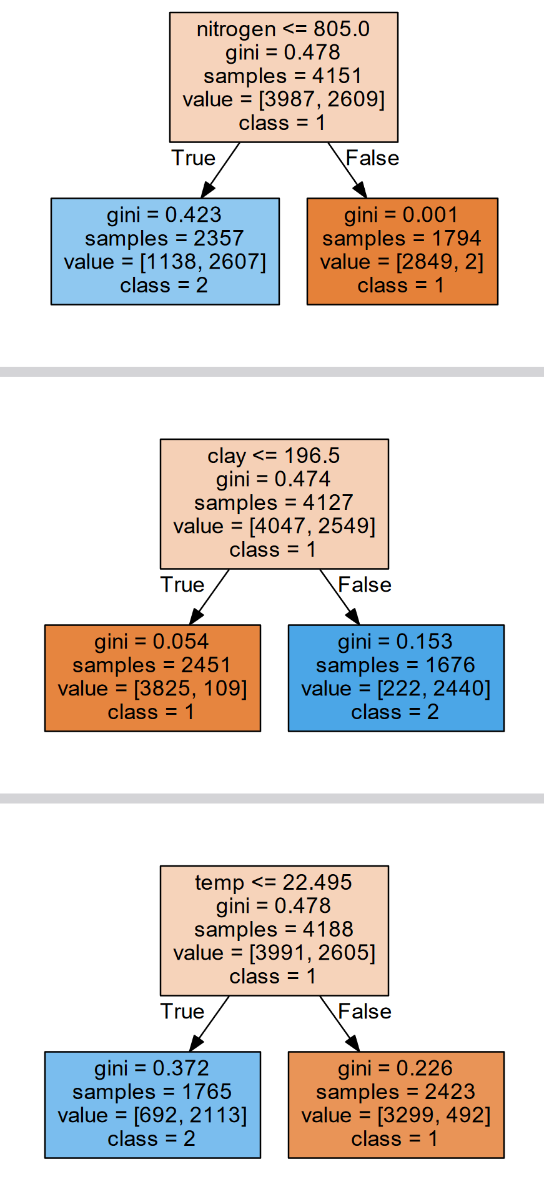
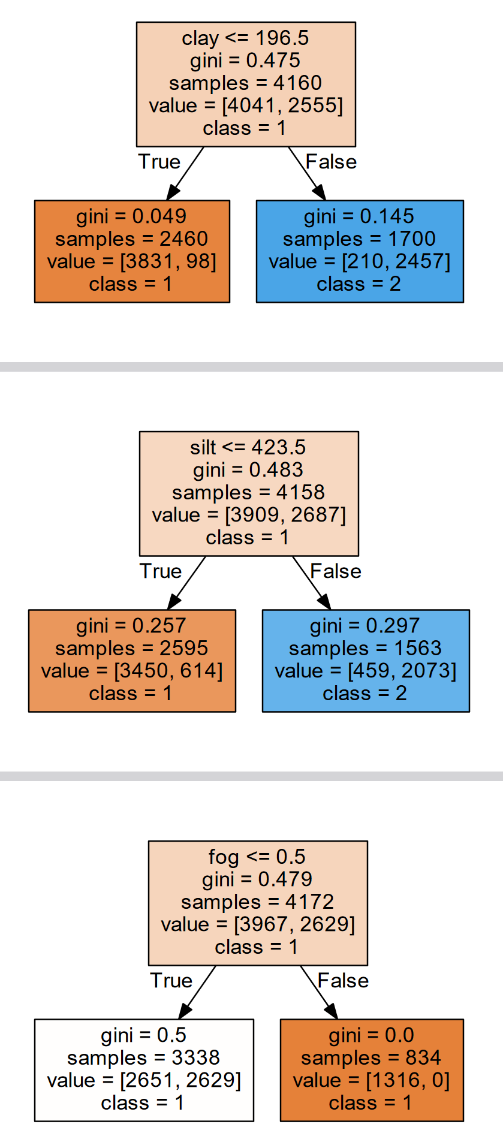
Обучение делится на несколько этапов:

1. Указываем классификатор в моем случае это RandomForestClassifier, подбираем диапазон параметров {'n\_estimators': range(10, 300, 10), 'max\_depth': range(1, 10)}. Max\_depth – максимльная глубина дерева, n\_estimators – число деревьев в лесу. Через GridSearchCV подбираем оптимальные параметры для обучения. В GridSearchCV экземпляре реализует обычный API оценки: когда «уместно» его на наборе данных всех возможные комбинации значений параметров оцениваются и наилучшее сочетание сохраняются. Оптимальные параметры для моей модели: {'max\_depth': 1, 'n\_estimators': 40}
2. В классификаторе вызваем метод обучение и передаем туда X\_train и y\_train. Результат леса:









# 7.3 Проверка обученных данных

**Кросс валидация**

Изучение параметров функции прогнозирования и тестирование ее на одних и тех же данных является методологической ошибкой: модель, которая будет просто повторять метки образцов, которые она только что увидела, будет иметь идеальную оценку, но пока не сможет предсказать что-либо полезное. невидимые данные. Такая ситуация называется **переобучением**. Чтобы этого избежать, при проведении (контролируемого) эксперимента с машинным обучением обычно используется часть имеющихся данных в виде **набора тестов** X\_test, y\_test.

При оценке различных настроек («гиперпараметров») для оценщиков, таких как Cнастройка, которая должна быть вручную установлена ​​для SVM, все еще существует риск переобучения на тестовом наборе, поскольку параметры можно настраивать до тех пор, пока оценщик не будет работать оптимально. Таким образом, знания о наборе тестов могут «просочиться» в модель, а показатели оценки больше не будут сообщать о производительности обобщения. Чтобы решить эту проблему, еще одна часть набора данных может быть представлена ​​как так называемый «набор для проверки»: обучение продолжается на обучающем наборе, после чего выполняется оценка на проверочном наборе, и когда эксперимент кажется успешным. , окончательную оценку можно провести на тестовом наборе.

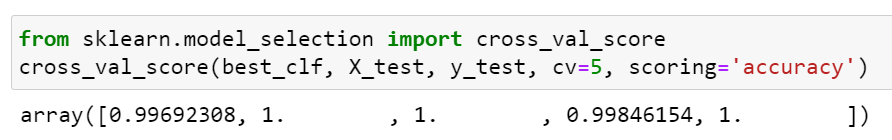
Однако, разбивая доступные данные на три набора, мы резко сокращаем количество выборок, которые можно использовать для обучения модели, а результаты могут зависеть от конкретного случайного выбора для пары наборов (обучение, проверка).

Решением этой проблемы является процедура [перекрестной проверки](https://en.wikipedia.org/wiki/Cross-validation_(statistics))  (cross-validation сокращенно CV). Набор тестов по-прежнему должен храниться для окончательной оценки, но набор для проверки больше не нужен при выполнении резюме. В базовом подходе, называемом k- кратным CV, обучающая выборка разбивается на k меньших наборов (другие подходы описаны ниже, но обычно следуют тем же принципам). Для каждой из k «фолдов» выполняется следующая процедура :

* Модель обучается с использованием �−1 складок в качестве обучающих данных;
* Результирующая модель проверяется на оставшейся части данных (т. е. она используется в качестве тестового набора для вычисления показателя производительности, такого как точность).

Показатель производительности, сообщаемый k- фолд перекрестной проверкой, тогда является средним из значений, вычисленных в цикле. Этот подход может быть дорогостоящим в вычислительном отношении, но не тратит слишком много данных (как в случае фиксации произвольного набора проверки), что является основным преимуществом в таких задачах, как обратный вывод, когда количество выборок очень мало.

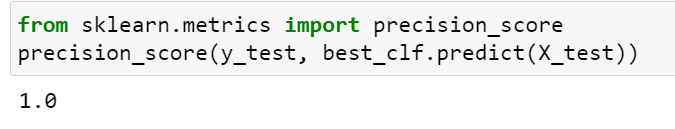
Значение точнсти у модели:



ROC

Точность — это отношение tp / (tp + fp), где tp — количество истинных срабатываний, а fp — количество ложных срабатываний. Точность — это интуитивно способность классификатора не маркировать отрицательный образец как положительный.

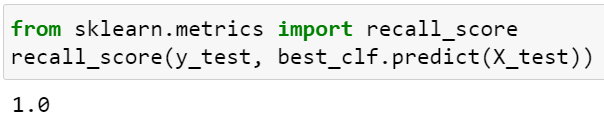
Лучшее значение равно 1, а худшее значение равно 0.



ROLL

Отзыв представляет собой отношение tp / (tp + fn), где tp — количество истинных положительных результатов, а fn — количество ложноотрицательных результатов. Под отзывом понимается интуитивно способность классификатора находить все положительные образцы.

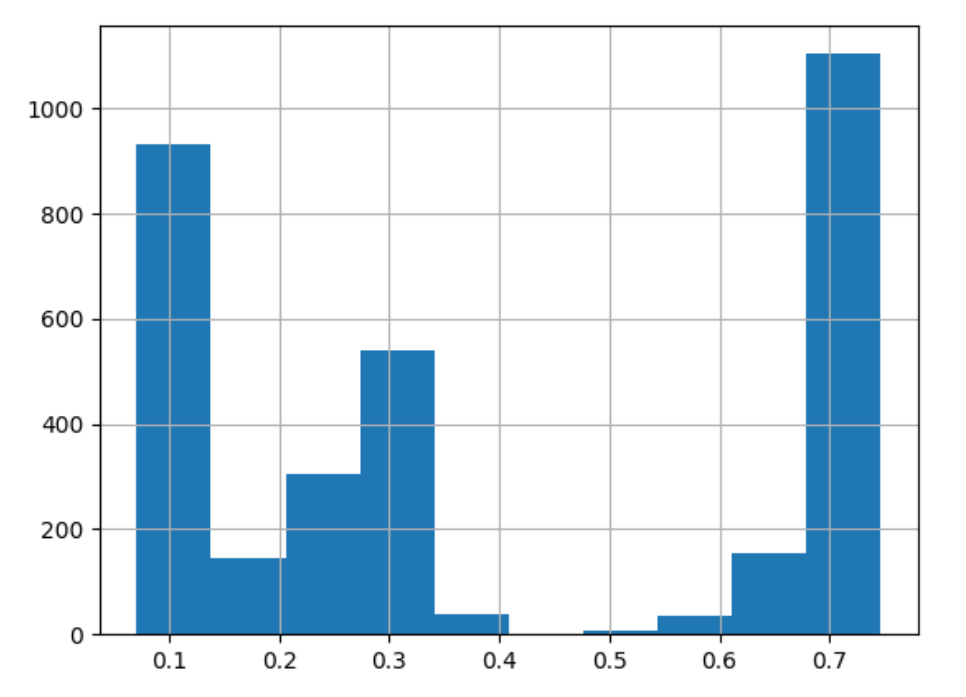
Лучшее значение равно 1, а худшее значение равно 0.



**Получение предсказание у тестовых данных**

В данном примере буду использовать тестовую выборку из X\_test

Передадим выборку в модель и получим вероятность принадлежности к классу 1(Пригодна для выращивания) для каждой локации из выборки, вот так распределились данные:

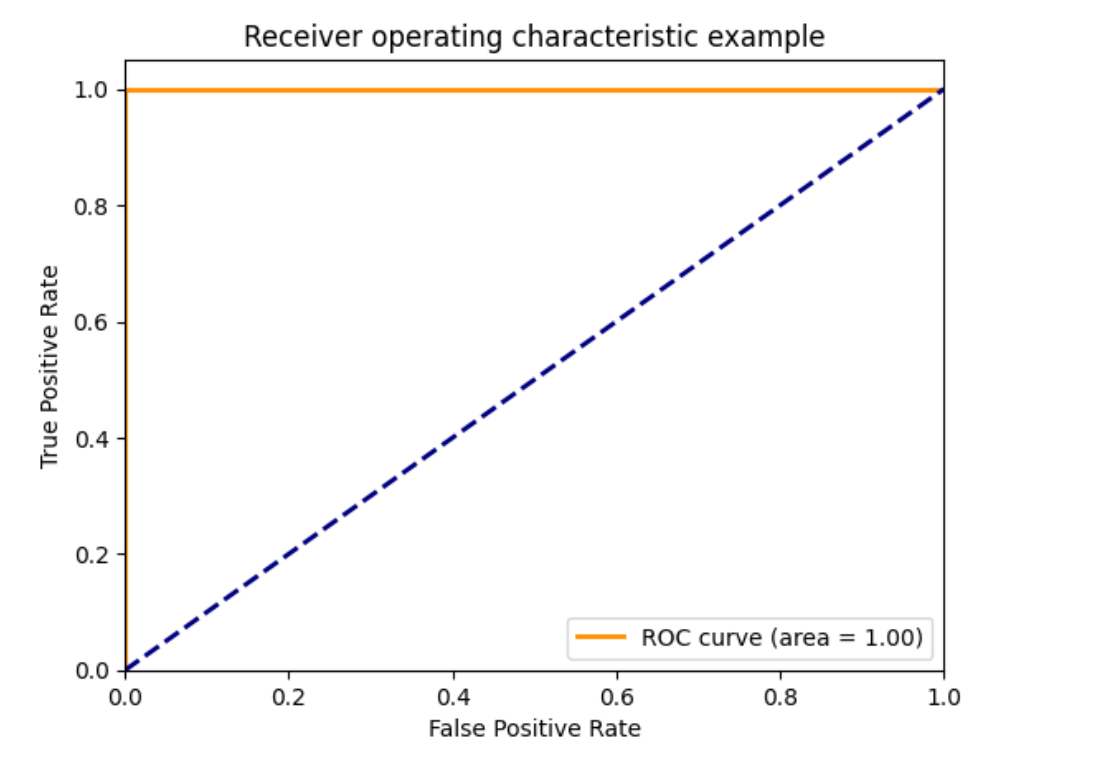


Слева кол-во локация, снизу вероятность

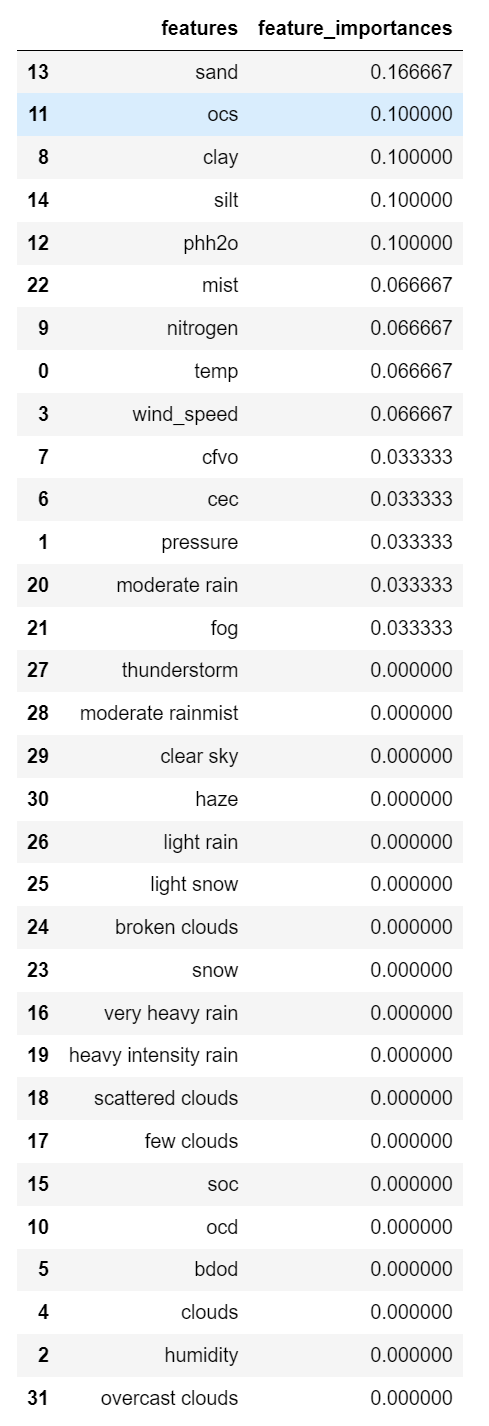
**рабочая характеристическая кривая приемника или кривая ROC**

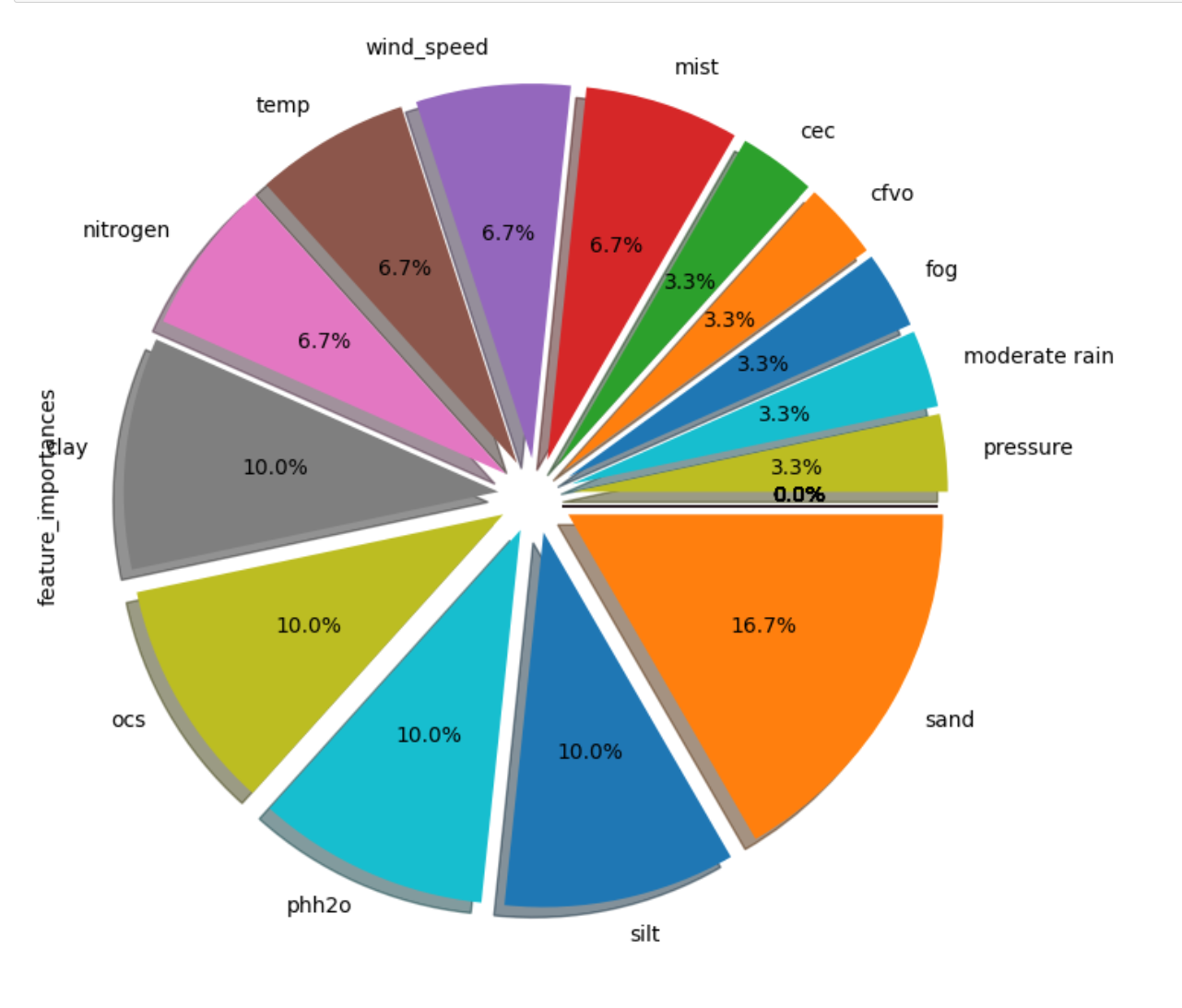
Кривую Roc будем вычислять через функцию roc\_curve.

Рабочая характеристика приемника (ROC), или просто кривая ROC, представляет собой графический график, который иллюстрирует работу системы двоичного классификатора при изменении ее порога дискриминации. Он создается путем построения графика доли истинных положительных результатов из положительных (TPR = частота истинных положительных результатов) по сравнению с долей ложных положительных результатов из отрицательных (FPR = частота ложных положительных результатов) при различных настройках пороговых значений. TPR также известен как чувствительность, а FPR — это единица минус специфичность или истинно отрицательный показатель.

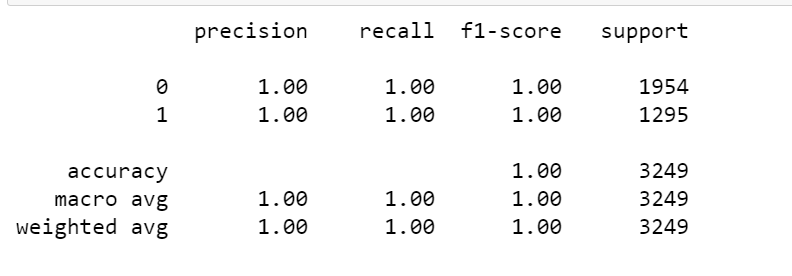


**Самые важные признаки по которым происходит определения**



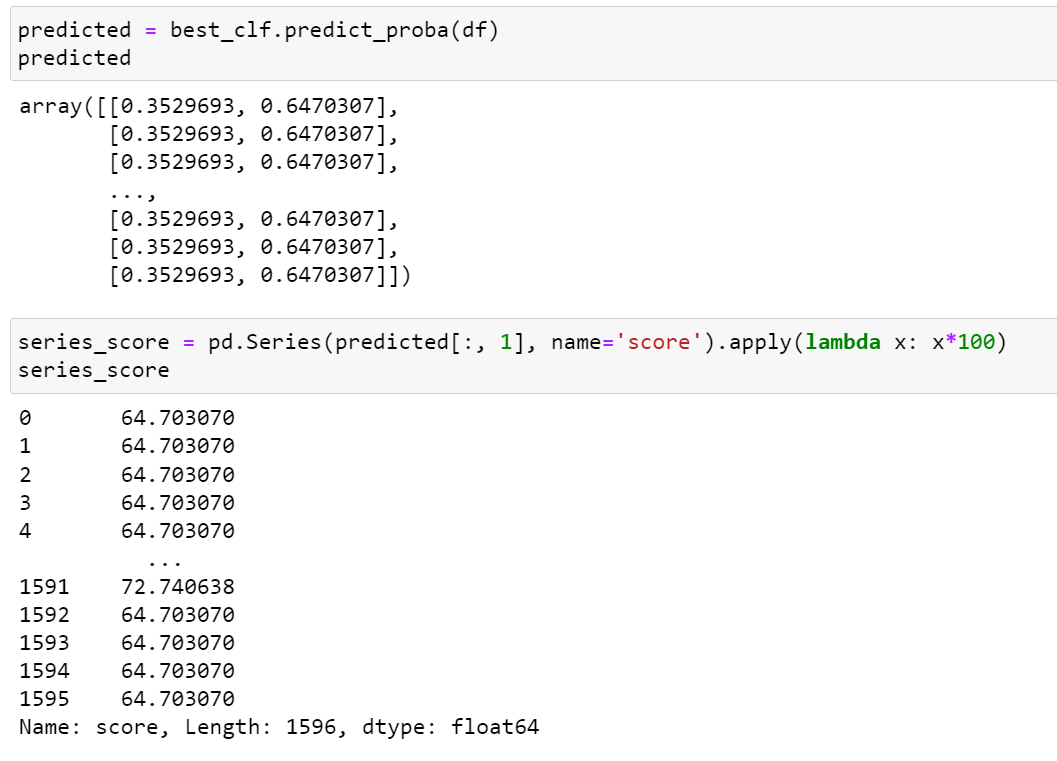


**Метрики оценки качества модели**



# 7.4 Получение предсказание

Получение предсказание состоит из нескольких этапов:

1. Передаем данные по которым хотим предсказать пригодность для выращивания
2. Делаем те же преобразование, что и для данных на обучения, приводим данные в модели читаемый формат
3. Отдельно сохраняем данные которые не используем в обучение, так как мы их вернем после получение скоринга
4. Получаем предикт и округляем его. 
5. Объединяем скоринг, данные которые не участвовали в получение предикта, данные которые участвовали в получение предикта
6. Прогоняем выборку через фичи которые изменяют скоринг в зависимости от критерия
7. Сохраняем данные

# 7.4.1 Фичи

**Фича наклон местности**

Если у местности большой наклон(45 градусов), то кофициент пригодности выставляется равным нулю

\* Скоринг – вероятность отнесение к классу где можно выращивать виноград

