

## **О повышении точности методов многоклассовой классификации на несбалансированных данных**

### ***Аннотация***

В работе исследованы различные методы извлечения наиболее информативных признаков в задачах многоклассовой классификации для несбалансированных и зашумленных данных. На примере статистических данных о заболеваниях кожи проведен их сравнительный анализ, показавший, что наиболее эффективным по точности классификации оказался алгоритм случайного леса при сэмплировании выборки с использованием алгоритма SMOTE.

**Ключевые слова.**

### ***Review***

Various methods of extracting the most informative features in multiclass classification problems for unbalanced and noisy data are explored. On the example of well known dermatology data set, their comparative analysis was carried out, which showed that the random forest algorithm was the most effective in terms of classification accuracy when sampling the data using the SMOTE algorithm.

### **1. ВВЕДЕНИЕ**

Задачи классификации являются одной из наиболее популярных в анализе данных [1]. Нередко возникают ситуации, когда в обучающем наборе данных доля примеров некоторого класса незначительна (этот класс будем называть миноритарным, а другой, преобладающий над первым, — мажоритарным). Такие тенденции хорошо заметны, например, в кредитном скоринге, медицине, маркетинге [2,3,4]. Построенный на таких наборах данных классификатор может оказаться абсолютно неэффективным, что приводит модель, обученную на таких данных, к переобучению и некорректным оценкам прогноза. Правильный выбор признаков может быть более значимой задачей, чем уменьшение времени обработки данных, или улучшения точности классификации. Следует отме-

тить, что могут отличаться и последствия ошибочной классификации. Причем неверная классификация примеров миноритарного класса, как правило, обходится в разы дороже, чем ошибочная классификация примера мажоритарного класса. К примеру, в медицине [5], нахождение минимального набора признаков, который является оптимальным для задачи классификации, может быть необходимым условием для разработки диагностического теста.

В настоящей работе проведены исследования методов преодоления разбалансированности классов, отбора наиболее информативно значимых признаков с целью повышения качества классификации и построения прогноза с точностью, более высокой, чем классическими алгоритмами классификации. На данных о заболеваниях кожи [6], проведены компьютерные эксперименты,...

Так как в последнее время метод опорных векторов (SVM) имеет отличные показатели по классификации и широко используется для диагностики заболеваний или медицинской помощи, мы будем рассматривать этот метод для исследования точности задач многоклассовой классификации для баз данных дерматологии. Для изучения точности классификации используются комбинации классификатора SVM и методов отбора различных наборов признаков RFE, Tree-based feature importance и Boruta с использованием балансирования классов методами RandomOverSampler, SMOTE, ADASYN и Borderline SMOTE. Экспериментальные результаты показывают, что точность классификации может быть более 98%. Перейдем к кратким теоретическим сведениям о наиболее распространенных стратегиях сэмплинга, а затем некоторые из них сравним, применив на наборе данных с несбалансированными классами.

## **2. Алгоритмы балансировки классов**

Одним из подходов для решения указанной проблемы является применение различных стратегий сэмплинга, которые можно разделить на две группы: случайные и специальные [5]. Восстановление баланса классов может проходить двумя путями. В первом случае удаляют некоторое количество примеров мажор-

ритарного класса (undersampling), во втором – увеличивают количество примеров миноритарного (oversampling).

## **2.1 Удаление примеров мажоритарного класса. Случайное удаление примеров мажоритарного класса (Random Undersampling)**

Для этого рассчитывается число  $K$  – количество мажоритарных примеров, которое необходимо удалить для достижения требуемого уровня соотношения различных классов. Затем случайным образом выбираются  $K$  мажоритарных примеров и удаляются. Примеры из мажоритарного класса могут удаляться не только случайным образом, но и по определенным правилам. Перейдем к рассмотрению таких стратегий.

## **Увеличение миноритарного класса. Дублирование примеров миноритарного класса (Oversampling). Случайная наивная выборка**

Самый простой способ увеличить количество примеров миноритарного класса — случайным образом выбрать наблюдения из него и добавить их в общий набор данных, пока не будет достигнут баланс между классами большинства и меньшинства. В зависимости от того, какое соотношение классов необходимо, выбирается количество случайных записей для дублирования. Одна из проблем со случайной наивной выборкой заключается в том, что она просто дублирует уже существующие данные. Достоинствами такого подхода являются его простота, лёгкость реализации и предоставляемая им возможность изменить баланс в любую нужную сторону. Про недостатки нужно говорить отдельно в соответствии с тем, какая стратегия сэмпинга используется: несмотря на то, что обе из них изменяют общий размер данных с целью поиска баланса, их применение имеет различные последствия. В случае undersampling удаление данных может привести к потере классом важной информации и, как следствие, понижения показателя его презентативности. В свою очередь применение oversampling может приводить к переобучению [6]. Такой подход к восстановлению баланса не всегда является эффективным, поэтому был предложен специальный метод увеличения числа примеров миноритарного класса – алгоритм SMOTE (Synthetic Minority Oversampling Technique) [7].

## Алгоритм SMOTE

Алгоритм SMOTE основан на идее генерации некоторого количества искусственных примеров, которые были бы «похожи» на имеющиеся в миноритарном классе, но при этом не дублировали их [8]. Для создания новой записи находят разность  $d = X_b - X_a$ , где  $X_a, X_b$  – векторы признаков «соседних» примеров  $a$  и  $b$  из миноритарного класса. Их находят, используя алгоритм ближайших соседей (*KNN*). В данном случае необходимо и достаточно для примера  $b$  получить набор из  $k$  соседей, из которого в дальнейшем будет выбрана запись  $b$ . Остальные шаги алгоритма *KNN* не требуются. Далее из  $d$  путем умножения каждого его элемента на случайное число в интервале  $(0, 1)$  получают  $\tilde{d}$ . Вектор признаков нового примера вычисляется путем сложения  $X_a$  и  $\tilde{d}$ . Алгоритм SMOTE позволяет задавать количество записей, которое необходимо искусственно сгенерировать. Степень сходства примеров  $a$  и  $b$  можно регулировать путем изменения значения  $k$  (числа ближайших соседей).

SMOTE решает многие проблемы, которые возникли у метода случайной выборки, и действительно увеличивает изначальный набор данных таким образом, что модель обучается гораздо эффективнее [9]. Тем не менее, данный алгоритм имеет и свои недостатки, главным из которых является игнорирование мажоритарного класса. Это может привести к тому, что при сильно разреженном распределении объектов миноритарного класса относительно мажоритарного наборы данных «смешаются», то есть расположатся в таком виде, что отделить объекты одного класса от другого будет очень трудно. Примером данного явления может служить случай, при котором между объектом и его соседом, на основе которых генерируется новый экземпляр, находится объект другого класса. В результате синтетически созданный объект будет находиться ближе к противоположному классу, чем к классу своих родителей. Кроме того, количество сгенерированных с помощью SMOTE экземпляров задаётся заранее, следовательно, уменьшается возможность изменения баланса и гибкость метода. Важно отметить существенное ограничение SMOTE. Поскольку он работает

путем интерполяции между редкими примерами, он может генерировать примеры только внутри тела доступных примеров—никогда снаружи. Формально SMOTE может только заполнить выпуклую оболочку существующих примеров меньшинства, но не создавать для них новые внешние области. Основное преимущество SMOTE по сравнению с традиционной случайной наивной чрезмерной выборкой заключается в том, что при создании синтетических наблюдений вместо повторного использования существующих наблюдений ваш классификатор с меньшей вероятностью будет перегружен. В то же время, всегда необходимо убедиться, что наблюдения, созданные SMOTE являются реалистичными.

### **Адаптивный синтетический сэмплинг и его обобщения**

В основе данного метода лежат алгоритмы синтетического сэмплинга. Основными из них являются Borderline-SMOTE и Adaptive Synthetic Sampling (ADASYN) [6]. Borderline-SMOTE накладывает ограничения на выбор объектов миноритарного класса, на основе которых генерируются новые экземпляры. Происходит это следующим образом: для каждого объекта миноритарного класса определяется набор  $k$  ближайших соседей, затем производится подсчёт, сколько экземпляров из этого набора принадлежит к мажоритарному классу (это число принимается за  $m$ ). После этого отбираются те объекты миноритарного класса, для которых верно неравенство  $k/2 \leq m < k$ . Полученный набор представляет собой экземпляры миноритарного класса, находящиеся на границе распределения, и именно у них вероятность оказаться некорректно классифицированными выше, чем у прочих. Следует отметить, почему неравенство, определяющее отбор объектов, исключает случаи, при которых все  $k$  соседей принадлежат мажоритарному классу: это связано с тем, что подобные экзем-

пляры расположены в зоне «смешивания» двух классов, и на их основе могут быть сгенерированы лишь искажающие процесс обучения модели объекты. В связи с этим они объявляются шумом (англ. noise) и игнорируются алгоритмом.

ADAZYN же, в свою очередь, основывается на систематическом методе, позволяющем адаптивно генерировать разные количества данных в соответствии с их распределениями [16]. Вначале рассчитывается количество объектов, которые необходимо сгенерировать для всего миноритарного класса, по формуле

$$G = (S_{major} - S_{minor}) * B,$$

где  $S_{minor}$  - выборка экземпляров миноритарного класса,  $B$  — параметр, используемый для определения желаемого уровня баланса [15]. Затем для каждого объекта миноритарного класса определяется  $k$  ближайших соседей в соответствии с Евклидовым расстоянием и высчитывается пропорция  $\Gamma$  по формуле  $\Gamma_i = \gamma_i / Z$ , где  $i$  принимает значение от 1 до количества экземпляров миноритарного класса,  $\gamma_i$  является числом  $k$  ближайших соседей объекта, которые принадлежат мажоритарному классу, а  $Z$  - константа нормализации, уравнивающая формулу ( $\Gamma_i$  - такая функция плотности распределения, что  $\sum \Gamma_i = 1$ ) [15]. Далее определяется количество экземпляров, которые необходимо сгенерировать для каждого объекта миноритарного класса:

**Input for the Algorithm: Training Dataset:  $D_r$  with  $m$  samples with  $\{\mathbf{x}_i, \mathbf{y}_i\}$ ,  $i = 1$  to  $m$ , where  $\mathbf{x}_i$  is an  $n$ -dimensional vector in feature space and  $\mathbf{y}_i$  is the corresponding class. Let  $\mathbf{m}_r$  and  $\mathbf{m}_x$  be the number of minority and majority class samples respectively, such that  $\mathbf{m}_r \leq \mathbf{m}_x$  and  $\mathbf{m}_r + \mathbf{m}_x = \mathbf{m}$**

## 2. Исследуемые данные: описание и характеристики

Дерматология - это медицинская наука, занимающаяся заболеваниями кожи. Часто поражения кожи и расстройства являются признаками внутренних болезней и отражают процессы, происходящие в организме. Кожа является самым

крупным органом в организме. Хотя обнаружить кожные заболевания, диагностировать симптомы и принять решение о терапии легче, чем для других системных заболеваниях, многие люди игнорируют их важность. Кожа является основным органом человеческого тела с целым рядом целей, которые поддерживают выживание. Диагностика эритематозно-плоскоклеточных заболеваний является серьезной проблемой в дерматологии, а современные принципы диагностики и лечения опираются на наиболее раннее обнаружение заболевания. Все они имеют общие клинические особенности с очень небольшими различиями. К заболеваниям этой группы относятся:

- Псориаз;
- Себорейный дерматит;
- Плоский лишай;
- Розовый лишай;
- Хронический дерматит;
- Красный волосяной лишай.

Еще одна трудность для диагностики заключается в том, что заболевание может проявлять признаки другого заболевания на начальной стадии и может иметь характерные признаки на последующих стадиях. В этом исследовании использовался набор данных дерматологии из Калифорнийского Университета в Ирвине (UCI) для проведения оптимизации параметров и оценки эффективности классификации с использованием линейной модели SVM [3]. Набор данных по дерматологии был создан компанией Nielsen в 1998 году и содержит 34 переменные признаков и 1 переменную класса (6 классов). Набор данных содержит 366 наблюдений. Классами являются псориаз (класс 1): 112 случаев; себорейный дерматит (класс 2): 72 случая; плоский лишай (класс 3): 61 случай; розовый лишай (класс 4): 49 случаев; хронический дерматит (класс 5): 52 случая и красный волосяной лишай (класс 6): 20 случаев. Поскольку заголовки занимают много места, мы провели кодирование атрибутов. Характеристика данных дерматологии приведены в Таблице 1. Набор данных дерматологии имеет восемь



пропусков. После их устранения, мы сохранили 358 наблюдений для данного исследования.

### **3. Компьютерные эксперименты**

Исследования данных проводились по следующей схеме:

1. Предварительная обработка базы данных: изменение пропусков в данных и использование кодирования атрибутов для набора данных. У нас осталось 358 экземпляров без изменения и 8 экземпляров с изменением пропусков в данных на среднее значение для дальнейшего эксперимента.
2. Балансирование классов с помощью сэмплинга.
3. Отбор признаков: исключение малоинформативных признаков. На этом этапе применялись следующие методы

#### ***Рекурсивное исключение признаков RFE***

Рекурсивное исключение признаков (RFE) итеративно удаляет признаки. Алгоритм начинается с полной модели SVM, содержащей все признаки, а затем удаляет наименее полезный предиктор в каждой итерации. Для более сложных задач это позволяет значительно сократить потребление памяти и ускорить вычисления. С другой стороны, нам также необходимо некоторое время выполнения самого выбора признаков, поскольку область поиска RFE имеет размер  $P$ , где  $P$  - число предикторов.

#### ***Выбор признаков на основе деревьев решений***

*Random Forest* алгоритм машинного обучения, предложенный в [7], представляющий собой ансамбль многочисленных, чувствительных к обучающей выборке алгоритмов (деревьев решений). Данные алгоритмы имеют малое смещение. Смещение (bias) метода обучения это отклонение среднего ответа обученного алгоритма от ответа идеального алгоритма. Каждый из этих классификаторов строится на случайном подмножестве объектов и случайном подмножестве признаков. Пусть обучающая выборка состоит из  $N$  примеров, размерность пространства признаков равна  $M$ , и задан



параметр  $m$ . Запишем пошагово алгоритм Random Forest.

Все деревья ансамбля строятся независимо друг от друга по следующей процедуре:

1. Сгенерируем случайную подвыборку с повторением размером  $n$  из обучающей выборки.
2. Построим решающее дерево, классифицирующее примеры данной подвыборки, причём в ходе создания очередного узла дерева будем выбирать признак, на основе которого производится разбиение, не из всех  $M$  признаков, а лишь из  $m$  случайно выбранных.
3. Дерево строится до полного исчерпания подвыборки и не подвергается процедуре прунинга (англ. pruning отсечение ветвей).

Классификация объектов проводится путём голосования: каждое дерево ансамбля относит классифицируемый объект к одному из классов, и побеждает класс, за который проголосовало наибольшее число деревьев.

Алгоритм Random Forest может быть использован в задаче оценки важности признаков. Для этого необходимо обучить алгоритм на выборке и во время построения модели для каждого элемента обучающей выборки посчитать out-of-bag-ошибку. Пусть  $X_n$  - бутстрапированная выборка дерева  $b_n$ . Бутстрэппинг представляет собой выбор  $l$  объектов из выборки с возвращением, в результате чего некоторые объекты выбираются несколько раз, а некоторые ни разу. Помещение нескольких копий одного объекта в бутстрапированную выборку соответствует выставлению веса при данном объекте соответствующее ему слагаемое несколько раз войдет в функционал, и поэтому штраф за ошибку на нем будет больше.

Затем для каждого объекта такая ошибка усредняется по всему случайному лесу. Чтобы оценить важность признака, его значения перемешиваются для всех объектов обучающей выборки и out-of-bag-ошибка считается снова. Важность признака оценивается путем усреднения по всем деревьям разности показателей out-of-bag-ошибок до и после перемешивания значений. При этом значения таких ошибок нормализуются на стандартное отклонение. Случайный лес имеет еще некоторые преимущества для использования его

в качестве алгоритма отбора признаков: он имеет очень мало настраиваемых параметров, относительно быстро и эффективно работает, что позволяет находить информативность признаков без значительных вычислительных затрат.

### ***Борута***

Эвристический алгоритм отбора значимых признаков, основанный на использовании Random Forest [5]. Суть алгоритма заключается в том, что на каждой итерации удаляются признаки, у которых  $Z$ -мера меньше максимальной  $Z$ -меры среди добавленных признаков. Чтобы получить  $Z$ -меру признака, необходимо посчитать важность признака, полученную с помощью встроенного алгоритма в Random Forest, и поделить ее на стандартное отклонение важности признака. Добавленные признаки получаются следующим образом: копируются признаки, имеющиеся в выборке, а затем каждый новый признак заполняется путем перетасовки его значений. В целях получения статистически значимых результатов эта процедура повторяется несколько раз, переменные генерируются независимо на каждой итерации.

Запишем пошагово алгоритм Boruta:

1. Добавить в данные копии всех признаков. В дальнейшем копии будем называть скрытыми признаками.
2. Случайным образом перемешать каждый скрытый признак.
3. Запустить Random Forest и получить  $Z$ -меру всех признаков.
4. Найти максимальную  $Z$ -меру из всех  $Z$ -мер для скрытых признаков.
5. Удалить признаки, у которых  $Z$ -мера меньше, чем найденная на предыдущем шаге.
6. Удалить все скрытые признаки.
7. Повторять все шаги до тех пор, пока  $Z$ -мера всех признаков не станет больше чем максимальная  $Z$ -мера скрытых признаков.

## **IV. Результаты исследования и их обсуждение**

Результаты исследования комбинаций моделей представлены в Таблице 2.

Сэмплинг для не-сбалансированных классов	Методы отбора признаков	Точность Теста	Количество выбранных признаков
На несбалансированных данных	Все признаки	0.9324	630
	RFE	0.9595	65
	Случайный лес	0.9595	32
	Борута	0.9324	207
Случайная выборка	Все признаки	0.9324	630
	RFE	0.9459	44
	Случайный лес	0.9565	44
	Борута	0.9595	284
SMOTE	Все признаки	0.9324	630
	RFE	0.9595	68
	Случайный лес	0.9595	42
	Борута	0.9459	257
ADASYN	Все признаки	0.9324	630
	RFE	0.9459	44
	Случайный лес	0.9595	40
	Борута	0.9459	276
Расширение SMOTE	Все признаки	0.9324	630
	RFE	0.9324	237
	<b>Случайный лес</b>	<b>0.9795</b>	42

	Борута	0.9595	238
--	--------	--------	-----

Показатели метода опорных векторов с использованием сбалансирования класса на основе случайной наивной чрезмерной выборки и выбором признаков на основе случайного леса обеспечила точность испытания 98,65%, что лучше, чем использование модели SVM на несбалансированных данных с точностью 93,24%.

### **Заключение.**

Отбор признаков является важным этапом построения алгоритмов машинного обучения. Данный этап необходим, чтобы избавиться от шумовых признаков и благодаря этому улучшить качество и ускорить работу алгоритмов. Проведенные вычислительные эксперименты подтверждают, что алгоритмы отбора признаков с помощью алгоритма случайного леса эффективно справляется со своей задачей

Работа с несбалансированными данными может быть чрезвычайно сложной задачей. Мы увидели на примере набора данных дерматологии как сбалансировать данные и выбрать признаки, используя несколько методов, и сравнили их результаты. Согласно экспериментальным результатам этого исследования, что касается влияния сэмплинга и выбора признаков на построение модели классификации SVM является огромным.

На практике применение сэмплинга помогло улучшить работу классификатора для некоторых алгоритмов по сравнению с обучением на несбалансированных данных [1]. Впрочем, то, насколько серьёзный прирост качества работы даст этот метод, зависит во многом от данных и задачи, которая стоит перед исследователем. К примеру, если допустить, что классификаторы могут обучаться и на несбалансированных данных, что, в принципе, не противоречит логике, есть вероятность получить результат, сопоставимый с применением того же алгоритма на данных, сбалансированных данным методом [11]. Тем не менее, для большинства несбалансированных данных применение сэмплинга действительно может помочь увеличить долю правильных решений классификатора.

Покупка или создание дополнительных данных

В качестве заключительного замечания, это сообщение в блоге было сосредоточено на ситуациях несбалансированных классов при молчаливом предположении, что вам дали несбалансированные данные, и вам просто нужно решить проблему дисбаланса. В некоторых случаях, как в конкурсе Kaggle, вам дают фиксированный набор данных, и вы не можете попросить больше. Но вы можете столкнуться с более сложной проблемой: у вас просто недостаточно примеров редкого класса. Ни один из методов выше, вероятно, не будет работать. Что ты делаешь?

В некоторых доменах реального мира вы можете купить или построить примеры редкого класса. Это область текущих исследований в области машинного обучения. Если редкие данные просто должны быть надежно помечены людьми, общий подход заключается в краудсорсинге их с помощью службы, такой как Mechanical Turk. Надежность человеческих меток может быть проблемой, но в машинном обучении была проделана работа по объединению человеческих меток для оптимизации надежности. Наконец, Клаудия Перлих в своем Strata talk All the Data and Still Not Enough приводит примеры того, как проблемы с редкими или несуществующими данными могут быть обработаны с помощью суррогатных переменных или проблем, по существу, используя прокси и скрытые переменные, чтобы сделать, казалось бы, невозможные проблемы возможными. С этим связана стратегия использования трансферного обучения для изучения одной проблемы и переноса результатов на другую проблему с редкими примерами.

## **Список литературы**

1. He H., Garcia A. Learning from Imbalanced Data // IEEE transactions on knowledge and data engineering, vol. 21, no. 9, September 2009. – P. 1263-1284.

2. Kursa M., Rudnicki W., "Feature Selection with the Boruta Package" *Journal of Statistical Software*, Vol. 36, Issue 11, Sep 2010.
3. N. V. Chawla, K. W. Bowyer, L. O'Hall, W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, 321–357, 2002.
4. He, Haibo, Yang Bai, Edwardo A. Garcia, and Shutao Li. "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," In *IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, 2008.
5. H. Han, W. Wen-Yuan, M. Bing-Huan, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," *Advances in intelligent computing*, 878–887, 2005
6. H. M. Nguyen, E. W. Cooper, K. Kamei, "Borderline over-sampling for imbalanced data classification," *International Journal of Knowledge Engineering and Soft Data Paradigms*, 3(1), pp.4–21, 2009.
7. <https://arxiv.org/pdf/1505.01658.pdf>
8. X. Lin, F. Yang, L. Zhou et al., "A support vector machine recursive feature elimination feature selection method based on artificial contrast variables and mutual information," *Journal of Chromatography B: Analytical Technologies in the Biomedical and Life Sciences*, vol. 10, pp. 149–155, 2012.
9. W. C. Hsu and T. Y. Yu, "Support vector machines parameter selection based on combined taguchi method and staelin method for e-mail spam filtering," *International Journal of Engineering and Technology Innovation*, vol. 2, no. 2, pp. 113–125, 2012.
10. [https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.over\\_sampling.ADASYN.html#imblearn.over\\_sampling.ADASYN](https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.over_sampling.ADASYN.html#imblearn.over_sampling.ADASYN).

11. X. Lin, F. Yang, L. Zhou et al., “A support vector machine recursive feature elimination feature selection method based on artificial contrast variables and mutual information,” *Journal of Chromatography B*, vol. 910, pp. 149–155, 2012.
12. Murphy, P. M. & Aha, D.W. (1994). UCI repository of machine learning databases. University of California-Irvine, Department of Information and Computer Science. <http://www1.ics.uci.edu/mllearn/> MLRepository.html.

**ТАБЛИЦА 1.** ПРИЗНАКИ БАЗЫ ДАННЫХ ДЕРМАТОЛОГИИ

ID	Атрибут
V1	Эритема
V2	Шелушение
V3	Определенные границы
V4	Зуд
V5	Феномен Кебнера
V6	Полигональные папулы
V7	Фолликулярные папулы
V8	Поражение слизистой оболочки полости рта
V9	Поражение колена и локтя
V10	Поражение кожи головы
V11	Семейный анамнез
V12	Недержание мочи меланином
V13	Эозинофилы в инфильтрате
V14	ПНЛ инфильтрат
V15	Фиброз папиллярной дермы
V16	Экзоцитоз
V17	Акантоз
V18	Гиперкератоз



V19	Паракератоз
V20	Забитость ребер гребня
V21	Удлинение ребер гребня
V22	Истончение супрапапиллярного эпидермиса
V23	Губчатая пустула
V24	мунро-микро-доступ
V25	Очаговый гипергранулез
V26	Исчезновение зернистого слоя
V27	Вакуолизация и повреждение базального слоя
V28	Спонгиоз
V29	Появление зубцов в виде зуба
V30	Заглушка фолликулярного рога
V31	Перифолликулярный паракератоз
V32	Воспалительный мононуклеарный инфилтрат
V33	Ленточный инфилтрат
V34	Возраст