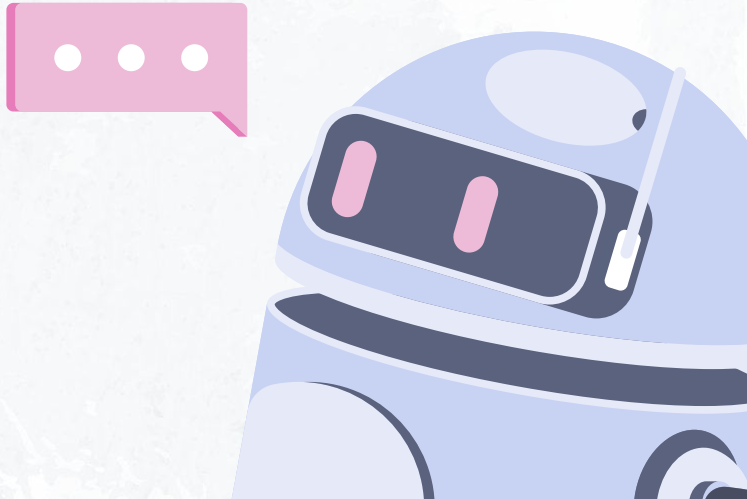
ПОЛИТЕХ
Санкт-Петербургский политехнический университет Петра Великого

Энкодинг категориальных данных



Подготовили ст.гр.5030102/10201:
Дмитриев Михаил
Хамидуллин Ильсаф

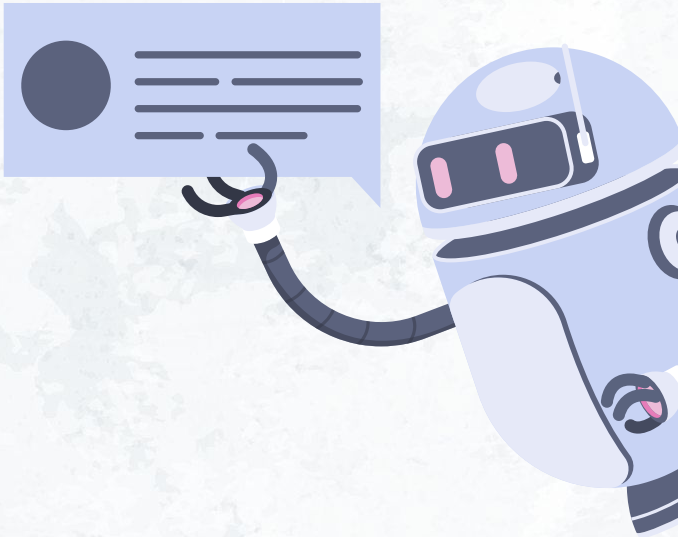
План выступления

01 → Введение

02 → Описание типов

03 → Код, реализация методов

04 → Заключние



01 →

Введение

Что, зачем и почему?

Энкодинг категориальных данных

– важный этап подготовки данных для задач машинного обучения, который напрямую влияет на эффективность и точность моделей.

В реальных данных часто встречаются **нечисловые**, **категориальные признаки**, такие как пол, профессия, место жительства, и т. д.

Эти данные, в отличие от числовых, не могут быть напрямую использованы многими алгоритмами машинного обучения, которые требуют **числового представления** для вычислений и анализа.

02 →

Методы кодирования категориальных данных

Разберем теорию с простыми примерами

01. One-Hot Encoding

(a) Описание ⟶

One-Hot Encoding создает новый столбец для каждого значения, где проставляется 1, если строка соответствует этому значению, и 0 в противном случае. Это наиболее распространенный метод для кодирования категориальных признаков.

(b) Пример ⟶

До кодирования: Цвет: Красный, Синий, Зелёный

После кодирования: Красный → [1, 0, 0] Синий → [0, 1, 0] Зелёный → [0, 0, 1]

02. Label Encoding

(a) Описание ⟶

Label Encoding присваивает уникальное целое число каждой категории, независимо от порядка. Этот метод лучше всего подходит для данных, где категориальные признаки можно представить числовыми метками.

(b) Пример ⟶

До кодирования: Цвет: Красный, Синий, Зелёный

После кодирования: Красный → 0 Синий → 1 Зелёный → 2

03. Ordinal Encoding

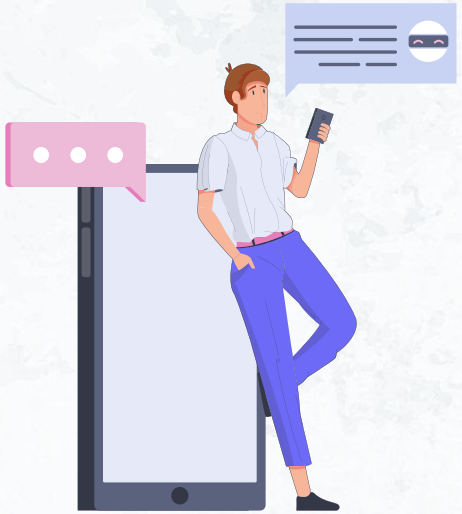
(a) Описание ⟶

Ordinal Encoding — присваивает каждому уникальному значению целое число в порядке их появления. Подходит для упорядоченных данных, таких как размер футболки (XS, S, M, L).

(b) Пример ⟶

До кодирования: Размер: XS, S, M

После кодирования: XS → 1, S → 2, M → 3



04. Frequency or Count Encoder

(a) Описание ⟶

Frequency Encoding заменяет каждую категорию её частотой или количеством появлений в данных. Подходит для категорий с разной частотой, например, названия городов.

(b) Пример ⟶

До кодирования: Город: Москва, Санкт-Петербург, Казань

После кодирования (частоты): Москва → 1000; Санкт-Петербург → 800; Казань → 500

05. Binary Encoding

(a) Описание

Binary Encoding комбинирует Label Encoding и бинарное представление чисел. Этот метод полезен для категорий с большим количеством уникальных значений, таких как ID.

(b) Пример

До кодирования: Москва, Санкт-Петербург, Казань

После кодирования: Москва → 001; Санкт-Петербург → 010; Казань → 011

06. Base-N Encoder

(a) Описание →

Base-N Encoding представляет категорию в произвольной системе счисления, например, в двоичной, троичной и т. д. Полезен для категорий с большим числом уникальных значений.

(b) Пример →

До кодирования: Категория: A, B, C

После кодирования (Base-3): A → 001; B → 002; C → 010

07. Helmert Encoding

(a) Описание ⟶

Helmert Encoding кодирует категорию как отклонение от среднего значения последующих категорий. Используется для анализа различных уровней категорий по сравнению с общей тенденцией.

(b) Пример ⟶

До кодирования: Категория: "Школьное", "Колледж", "Университет"

После кодирования: "Школьное" → [-1, -1]; "Колледж" → [1, -1]; "Университет" → [0, 2]

08. Mean Encoding или Target Encoding

(a) Описание ⟶

Mean Encoding заменяет категории на среднее значение целевой переменной для каждой категории. Часто используется в задачах предсказания.

(b) Пример ⟶

До кодирования: Город: Москва, Санкт-Петербург

Средняя цена недвижимости: Москва → 15, Санкт-Петербург → 20

После кодирования: Москва → 15 Санкт-Петербург → 20

09. Weight of Evidence Encoding

(a) Описание ⟶

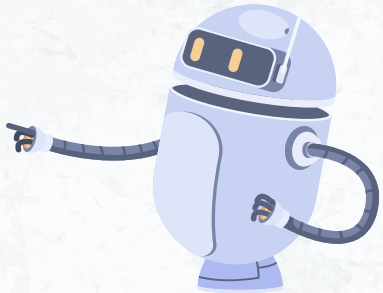
Weight of Evidence Encoding вычисляет логарифмическое отношение вероятностей каждой категории к целевому значению. Подходит для бинарных целевых переменных.

(b) Пример ⟶

До кодирования: Клиенты: Платят, Не платят

После кодирования (WOE):

Платят → log(P(Платят) / P(Не платят))
Не платят → log(P(Не платят) / P(Платят))



10. Sum Encoder (Deviation Encoding или Effect Encoding)

(a) Описание ⟶

Sum Encoding кодирует категории на основе отклонения от среднего значения всех категорий. Часто используется в линейных моделях.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → [-1, 1] B → [1, -1] C → [0, 0]

11. Leave-one-out Encoder (LOO или LOOE)

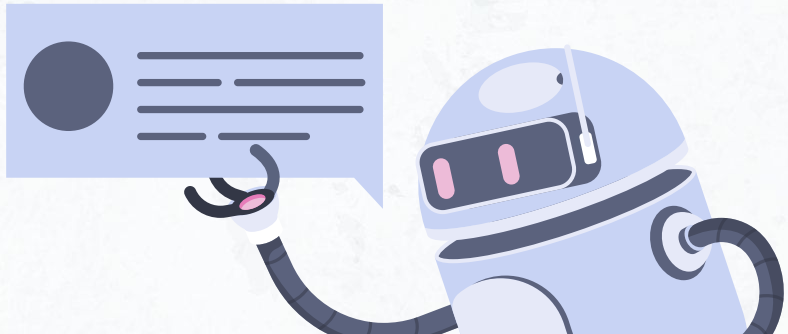
(a) Описание ⟶

Leave-one-out Encoding аналогичен Target Encoding, но исключает текущее наблюдение при расчёте среднего. Это помогает избежать переобучения.

(b) Пример ⟶

До кодирования: Категория: A = 70%, B = 50%, C = 40%

После кодирования: A → 0.7, B → 0.5, C → 0.4



12. CatBoost Encoder

(a) Описание ⟶

CatBoost Encoder — это метод, встроенный в алгоритм CatBoost, оптимизированный для категориальных данных. Уменьшает вероятность переобучения.

(b) Пример ⟶

До кодирования: Категория: A = 1, B = 0, A = 1, C = 0, B = 1

После кодирования:

Для категории "A" среднее значение целевой переменной: (1 + 1) / 2 = 0.67
Для категории "B" среднее значение: (0 + 1) / 2 = 0.50
Для категории "C" значение: 0 / 1 = 0.00

13. James-Stein Encoding

(a) Описание ⟶

James-Stein Encoding использует комбинацию категории и среднего по всей выборке, уменьшая переобучение. Полезен для малых данных.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → комбинация уникального и общего среднего

14. M-estimator Encoding

(a) Описание ⟶

M-estimator Encoding — метод сглаживания, использующий параметр для учета малых категорий. Полезен для данных с редкими категориями.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → (среднее по A * k + общее среднее) / (k + 1)

15. Hashing Encoding

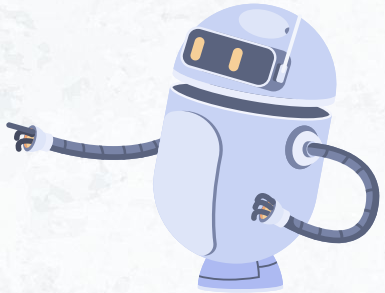
(a) Описание ⟶

Hashing Encoding использует хеш-функцию для распределения категорий в фиксированное количество колонок. Экономит пространство, подходит для больших наборов данных.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → хэш1; B → хэш2; C → хэш1



16. Backward Difference Encoding

(a) Описание ⟶

Backward Difference Encoding кодирует каждую категорию, сравнивая её с последующей. Используется для анализа трендов.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → сравнение с B и C B → сравнение с C

17. Polynomial Encoding

(a) Описание ⟶

Polynomial Encoding расширяет Sum Encoding, добавляя полиномиальные признаки. Используется для анализа нелинейных зависимостей.

(b) Пример ⟶

До кодирования: Категория: A, B, C

После кодирования: A → [1, x] B → [1, x^2] C → [1, x^3]

18. MultiLabelBinarizer

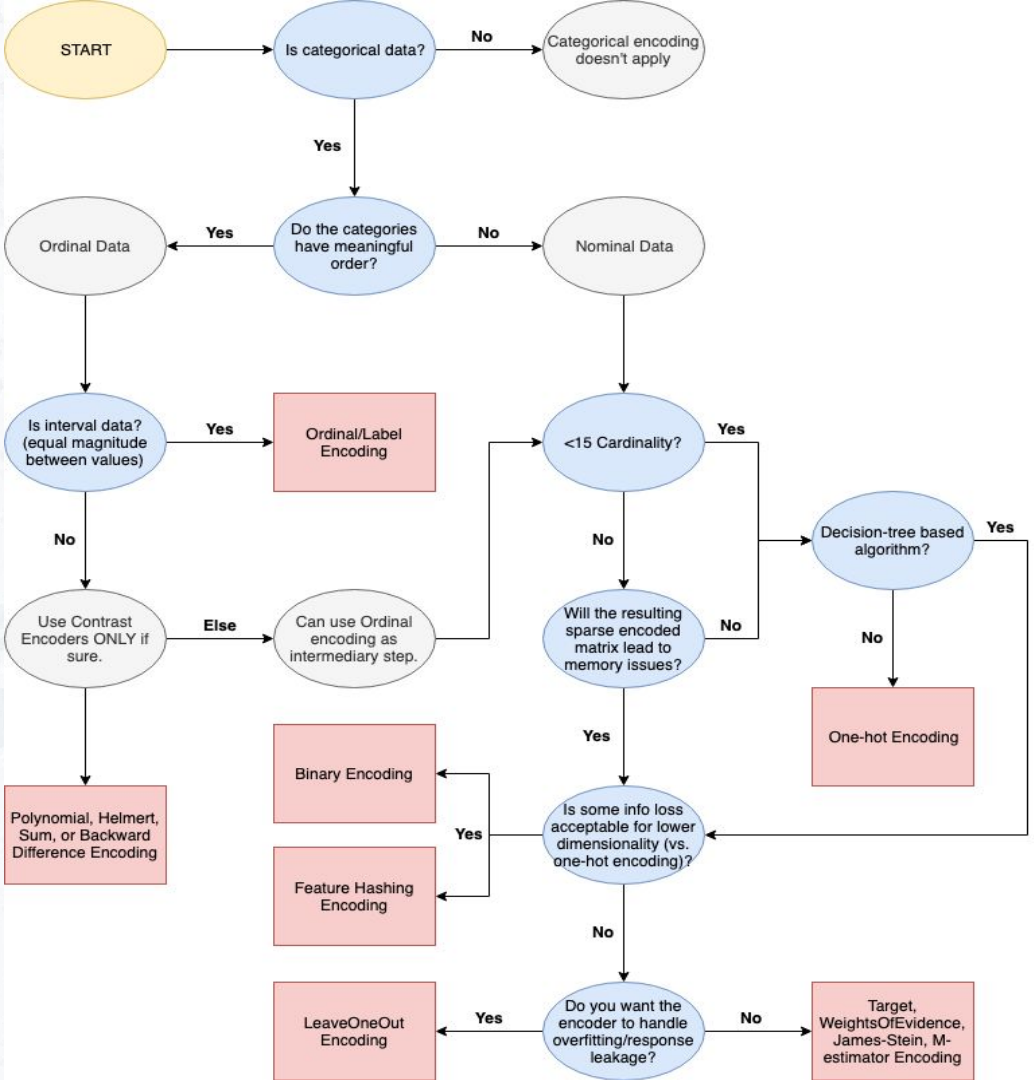
(a) Описание ⟶

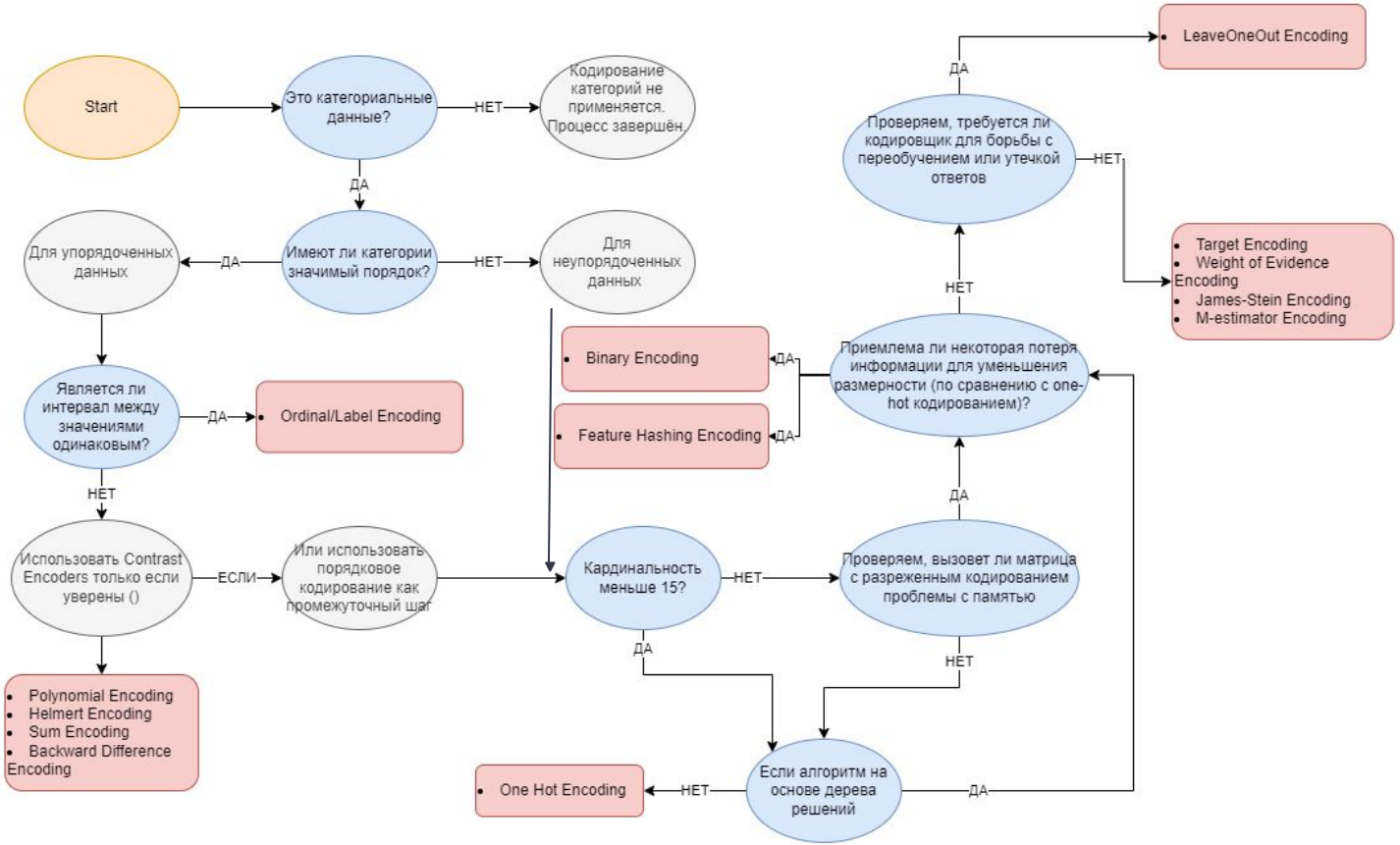
MultiLabelBinarizer преобразует многозначные категории в бинарные столбцы. Полезен для работы с категориями с множественным выбором.

(b) Пример ⟶

До кодирования: ['A', 'B'], ['B'], ['A', 'C'], ['C', 'D']

После кодирования: [[1, 1, 0, 0], [0, 1, 0, 0], [1, 0, 1, 0], [0, 0, 1, 1]]


START
Is categorical data?
No
Categorical encoding doesn't apply
Yes
Do the categories have meaningful order?
Yes
Ordinal Data
No
Nominal Data
Is interval data? (equal magnitude between values)
Yes
Ordinal/Label Encoding
No
Use Contrast Encoders ONLY if sure.
Else
Can use Ordinal encoding as intermediary step.
Polynomial, Helmert, Sum, or Backward Difference Encoding
<15 Cardinality?
Yes
No
Will the resulting sparse encoded matrix lead to memory issues?
No
Yes
Decision-tree based algorithm?
Yes
No
One-hot Encoding
Yes
Binary Encoding
Feature Hashing Encoding
Is some info loss acceptable for lower dimensionality (vs. one-hot encoding)?
Yes
No
LeaveOneOut Encoding
Yes
Do you want the encoder to handle overfitting/response leakage?
No
Target, WeightsOfEvidence, James-Stein, M-estimator Encoding


Start
Это категориальные данные?
НЕТ
Кодирование категорий не применяется. Процесс завершён.
ДА
Имеют ли категории значимый порядок?
ДА
Для упорядоченных данных
НЕТ
Для неупорядоченных данных
Является ли интервал между значениями одинаковым?
ДА
Ordinal/Label Encoding
НЕТ
Использовать Contrast Encoders только если уверены ()
ЕСЛИ
Или использовать порядковое кодирование как промежуточный шаг
Polynomial Encoding
Helmert Encoding
Sum Encoding
Backward Difference Encoding
Кардинальность меньше 15?
НЕТ
ДА
Проверяем, вызовет ли матрица с разреженным кодированием проблемы с памятью
НЕТ
ДА
Приемлема ли некоторая потеря информации для уменьшения размерности (по сравнению с one-hot кодированием)?
ДА
Binary Encoding
ДА
Feature Hashing Encoding
ДА
Проверяем, требуется ли кодировщик для борьбы с переобучением или утечкой ответов
НЕТ
ДА
LeaveOneOut Encoding
НЕТ
Target Encoding
Weight of Evidence Encoding
James-Stein Encoding
M-estimator Encoding
Если алгоритм на основе дерева решений
НЕТ
One Hot Encoding
ДА

03 →

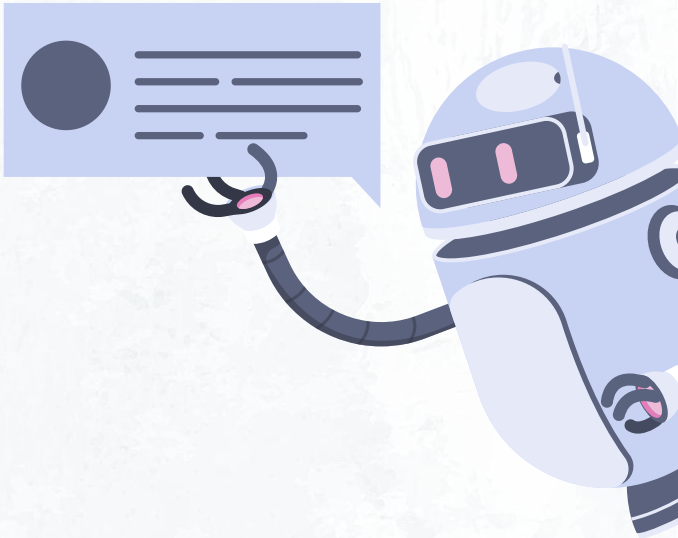
Код - реализация методов

Смотри на: https://github.com/Ilsaffff/categorial-data-encoding

04 →

Заключение

К чему мы пришли?

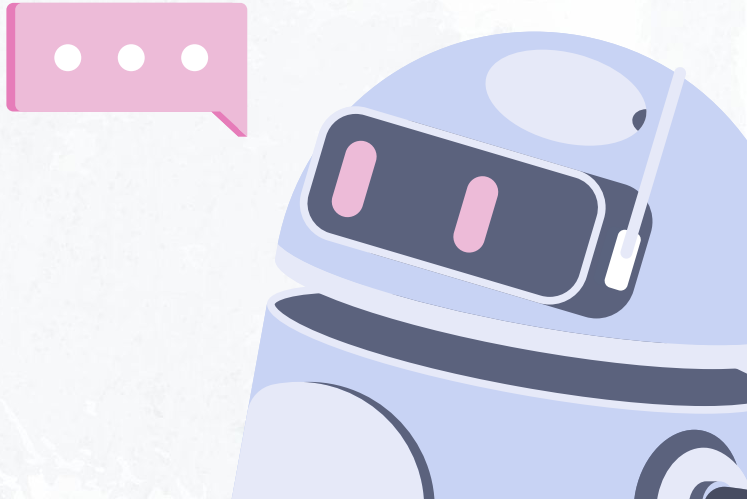


Источники

- https://medium.com/aiskunks/categorical-data-encoding-techniques-d6296697a40f
- https://github.com/alteryx/categorical_encoding
- https://github.com/scikit-learn-contrib/category_encoders

ПОЛИТЕХ
Санкт-Петербургский политехнический университет Петра Великого

Энкодинг категориальных данных



Подготовили ст.гр.5030102/10201:
Дмитриев Михаил
Хамидуллин Ильсаф