

# Отчет по этапу №5 индивидуального проекта

## Операционные системы

---

Уржиндорж Мягмар

31 августа 2024

Российский университет дружбы народов, Москва, Россия

- Уржиндорж Мягмар
- Студент, 1 курс
- Факультет физико-математических и естественных наук
- Российский университет дружбы народов
- 1032235131@pfur.ru
- <https://muagmar-urzhindorzh.github.io/>

Целью данной лабораторной работы является приобретение навыков по добавлению информации о себе на сайт научного сотрудника и выполнение 5 этапа индивидуального проекта.

# Выполнение лабораторной работы

Создаю записи по своему проекту на тему “AI\_Arrow”, редактируя файл `index.md`, находящийся в папке по адресу `/home/muagmar-urzhindorzh/work/blog/content/project`.

...

## ### Категориальные фичи

Категориальные фичи обрабатываются с помощью `LabelEncoder`, который кодирует категориальные признаки, присваивая каждой уникальной категории числовое значение.

Как работает `LabelEncoder`?

`LabelEncoder` присваивает каждой уникальной категории числовую метку (`label`). Процесс преобразования состоит из двух этапов:

Обучение (`fit`):

`LabelEncoder` анализирует данные и выявляет все уникальные категории, создавая для них числовые метки.

Например, для категорий `['cat', 'dog', 'fish']` метки могут быть присвоены следующим образом:

`cat` → 0

`dog` → 1

`fish` → 2

Преобразование (`transform`):

После обучения `LabelEncoder` может преобразовать исходные категориальные данные в их числовые метки.

- `NumOfProducts`: количество разных продуктов банка у пользователя (таких как сберегательный счёт, кредитная карта и др.)
- `HasCrCard`: есть ли у пользователя кредитная карта
- `IsActiveMember`: активный ли пользователь банка
- `EstimatedSalary`: зарплата пользователя

В результате мы хотим предсказать:

- `Exited`: Whether the customer has churned (1 = yes, 0 = no)

*# Посмотрим, сколько у нас данных*

```
num_train_rows, num_train_columns = train_data.shape
```

```
num_test_rows, num_test_columns = test_data.shape
```

```
print("Training Data:")
print(f"Количество строк: {num_train_rows}")
print(f"Количество столбцов: {num_train_columns}\n")
```

```
print("Test Data:")
print(f"Количество строк: {num_test_rows}")
print(f"Количество столбцов: {num_test_columns}\n")
```

# Выполнение лабораторной работы

## Пишу пост по прошедшей неделе.

после посещения шанхая я отправился в пекин, чтобы изучить его знаменитые достопримечательности. В первую очередь я решил посетить Запретный город, известный своим богатым историческим наследием и архитектурой. Прогуливаясь по его территории, я ощутил атмосферу древности и величия китайской культуры.

```

```

Далее я отправился в Храм Неба, который славится своей красотой и гармоничным сочетанием природы и архитектуры. Здесь я наблюдал за церемониями поклонения небу, которые проводятся каждый год. Следующим пунктом моего маршрута стал Летний дворец, расположенный на окраине Пекина. Этот парк поражает своей красотой и разнообразием ландшафтов, здесь можно отдохнуть от городской суеты и насладиться природой.

```

```

Также я посетил Великую китайскую стену, которая является символом могущества и величия китайской цивилизации. Прогулка по этому грандиозному сооружению позволила мне ощутить масштаб и величие этого исторического памятника.

```

```

В свободное время я изучал местные рынки и магазины, где приобрёл сувениры и подарки для своих близких.

В заключение своего визита в Пекин я посетил площадь Тяньаньмэнь, где находится мавзолей Мао Цзэдуна и другие важные исторические объекты. Здесь я ощутил дух китайской нации и понял, насколько важно сохранять и уважать свою историю.

```

```

Моё путешествие в Пекин оставило неизгладимые впечатления и позволило мне глубже погрузиться в

## Как прошла моя неделя 4

Aug 20, 2024 · 🌐 Мягмар Уржиндорж · 1 min read



После посещения Шанхая я отправился в Пекин, чтобы изучить его знаменитые

Рис. 2: Добавление поста по прошедшей неделе

# Выполнение лабораторной работы

## Добавляю пост на тему “Языки научного программирования”.

слайдов.

### ## Язык программирования Julia.

Язык **Julia** – это кросс-платформенный компилируемый свободно распространяемый язык программирования (лицензия **MPL**) с динамической типизацией, который имеет ряд достоинств и недостатков.

В **Julia** реализована возможность **JIT** – компиляции на основе **LLVM**. Компиляция **Just-in-Time (JIT)** позволяет обеспечить одновременно выразительность современных интерпретируемых языков и производительность таких языков, как **C** и **Fortran**. Компилятор **JIT** выполняет компиляцию во время первого запуска программы, извлекая из текста информацию, не указанную явно программистом, и используя эту информацию для оптимизации создаваемого машинного кода.

К достоинствам языка **Julia** можно отнести следующее:

- Простота интуитивно понятный язык, синтаксис которого напоминает синтаксис **Python** и **MATLAB**;
- быстроедействие вычислений программ, написанных на **Julia**, сопоставимо с быстроедствием программ, написанных на **C** или **Fortran**;
- эффективная **векторизация** и распараллеливание вычислений, большое число типов данных, включая рациональный и комплексные числа;
- проведение расчетов при отсутствии некоторых данных (есть тип дан **NaN missing**);
- задание точности вычислений (тип данных **BigFloat**, функция **setprecision**);
- реализация символьных вычислений с помощью **Macro** команд;
- использование библиотек, написанных на **C** и **Fortran**, и обмен библиотеками с **Python** и **R**;
- интеграция с СУБД (**PostgreSQL**, **MySQL**, **JSON**);
- поддержка символов **Unicode**;
- использование парадигмы множественной диспетчеризации – вызов функции слабо зависит от типа параметров функции (параметрический полиморфизм);
- наличие хорошей встроенной математической библиотеки с функциями линейной алгебры.

## Языки научного программирования

Aug 20, 2024 · Мягмар Уржиндорж · 4 min read

```
152     changePhotoDescription(cell) {
153     }
154
155     function updatePhotoDescription() {
156     if (descriptions.length > (page * 9) + (currentImage.substring(0, 9)))
157     document.getElementById('bigImageDesc').innerHTML = descriptions[page * 9 + currentImage.substring(0, 9)]
158     }
159
160     function updateAllImages() {
161     var i = 1;
162     while (i < 10) {
163     var elementId = 'foto' + i;
164     var elementIdBig = 'bigImage' + i;
165     if (page * 9 + i - 1 < photos.length) {
166     document.getElementById( elementId ).src = 'images/' + photos[page * 9 + i - 1] + '.jpg';
167     document.getElementById( elementIdBig ).src = 'images/' + photos[page * 9 + i - 1] + '.jpg';
168     } else {
169     document.getElementById( elementId ).src = '';
170     }
```

Рис. 3: Добавление поста по теме

В ходе выполнения данной лабораторной работы я приобрел навыки по добавлению информации о себе на сайт научного сотрудника и выполнил 5 этап индивидуального проекта.