

# Содержание

<b>Аннотация</b>	<b>4</b>
<b>1 Введение</b>	<b>5</b>
1.1 Описание предметной области . . . . .	5
1.2 Постановка задачи . . . . .	5
1.3 Структура работы . . . . .	6
<b>2 Обзор литературы</b>	<b>7</b>
2.1 Метрическое и контрастивное обучение . . . . .	7
2.2 Инвариантные модели . . . . .	8
2.2.1 SimCLR . . . . .	8
2.2.2 Redundancy reduction. Barlow Twins . . . . .	9
2.2.3 Siamese networks. SimSiam . . . . .	9
2.3 Эквивариантные модели . . . . .	10
2.3.1 EquiMod . . . . .	10
2.3.2 E-SSL . . . . .	11
<b>3 Анализ пространств признаков и эмбэдингов</b>	<b>11</b>
3.1 Задачи анализа . . . . .	12
3.2 Метрики, используемые при анализе . . . . .	12
3.3 Архитектурные составляющие . . . . .	13
<b>4 Экспериментальное исследование</b>	<b>13</b>
4.1 Визуализация пространств . . . . .	13
4.1.1 SimCLR . . . . .	13
4.1.2 Barlow Twins . . . . .	14
4.1.3 SimSiam . . . . .	15
4.1.4 E-SSL . . . . .	15
4.2 Геометрическое и топологическое сравнение . . . . .	16
4.2.1 Оценка внутренней размерности . . . . .	16
4.2.2 Топологическая оценка сходства пространств . . . . .	17
4.3 Точность предсказаний, knn . . . . .	17

<b>5 Выводы</b>	<b>18</b>
5.1 Результаты . . . . .	18
5.2 Перспективы дальнейшей работы . . . . .	18
<b>Список литературы</b>	<b>19</b>

## **Аннотация**

В данном исследовательском проекте рассмотрены современные подходы к метрическому обучению без учителя. Большое внимание уделено изучению контрастивных и эквивариантных методов обучения, выявлению их общих принципов и отличий. Одна из главных задач работы – реализация изученных методов для исследования пространства репрезентаций (признаков) и пространства вложений (эмбэдингов). Основным акцентом исследования является упрощенный топологический и геометрический анализ пространств, построенных во время обучения моделей.

## **Ключевые слова**

Глубокое обучение, метрическое машинное обучение, обучение без учителя, контрастивность, инвариантность, эквивариантность, пространство признаков, пространство вложений.

# 1 Введение

## 1.1 Описание предметной области

Многие задачи компьютерного зрения, обработка естественного языка сталкиваются с неразмеченными данными, которые трудоемко классифицировать вручную. В связи с этим в последнее время развивается SSL-подход, который позволяет изучать сложные паттерны неразмеченного датасета.

Метрический подход, при котором обучение происходит за счет минимизации "расстояния" между эмбедингами, зачастую применяется в SSL. Одним из его примеров является контрастивное обучение – модель обучается как по принципу "близости" семантически одинаковых объектов, полученных с помощью аугментации исходных данных, так и по принципу различия. Таким образом, модель становится инвариантной к трансформациям. Однако для некоторых задач чувствительность к определенным трансформациям играет ключевую роль [10], поэтому эквивариантные методы становятся более актуальными.

Многие инвариантные и эквивариантные современные методы используют несколько представлений входных данных во время обучения. Кроме основной нейронной сети, строящей пространство признаков, используются различные предикторы и проекторы для получения векторов меньшей размерности. Таким образом, во время обучения модель формирует несколько отличающихся друг от друга представлений одних и тех же данных.

## 1.2 Постановка задачи

В данной работе изложены основные концепции метрического и контрастивного обучения, разобраны базовые инвариантные модели: SimCLR [3], Barlow Twins [11], SimSiam [4]. Также в работе представлен анализ статей [5, 6], выявлены характерные черты эквивариантного подхода, отличия от инвариантного, разобраны модели E-SSL и EquiMod.

В процессе исследования протестирован каждый подход, построена визуализация пространства признаков и пространства вложений всех моделей, рассмотрена геометрическая и топологическая составляющая полученных пространств. Задачей проекта является выявление особенностей пространств репрезентаций и вложений для инвариантных и эквивариантных моделей. Кроме того, в работе эмперически показана необходимость использования векторов признаков в качестве выхода модели, а не эмбеддингов для каждого метода.

### 1.3 Структура работы

Работа организована следующим образом. В разделе 2 дается обзор релевантной литературы. Далее, в разделе 3 описывается план анализа, его технические составляющие. В подразделе 4.1 представлена визуализация пространств и некоторые численные характеристики. В последующих подразделах использованы более сложные методы анализа. Конечные результаты данного исследования и выводы можно найти в разделе 5.

## 2 Обзор литературы

### 2.1 Метрическое и контрастивное обучение

Парадигма метрического обучения строится вокруг существования функции от двух или более объектов, оценивающей расстояние (похожесть) между ними. Реализация подхода есть нейронная сеть, в которой модель тренируется переводить изображения в многомерное пространство признаков, минимизируя функцию потерь, измеренную на полученных объектах. Например, базовый метод метод  $k$  ближайших соседей использует метрику для определения, к какому классу должен принадлежать целевой объект выборки.

Однако у таких методов есть существенные недостатки, отмеченные в статье [8]. Во-первых, изначально предполагается, что входное пространство подчиняется какой-то метрике. Во-вторых, в процессе обучения модель не строит явно функцию, которая может отображать входные данные, отличающиеся от исходных, тем самым, она не всегда корректно работает на датасете другой природы. Авторы этой же статьи предлагают контрастивный подход, схожий с пружинной системой. Изначально для каждого входного вектора  $X$  создается множество  $S_X$  из векторов, "похожих" на  $X$ . Авторы предлагают использовать некоторую предварительную информацию для формирования  $S_X$ . Затем модель строит функцию  $G_W$ , отображающую исходные данные в пространство признаков так, чтобы эмбединги векторов из  $S_X$  располагались ближе к  $X$ , чем эмбединги векторов из  $\overline{S_X}$ . Такой эффект достигается при минимизации функции:

$$L(W, Y, X_1, X_2) = \frac{1}{2}(1 - Y)D_W^2 + \frac{1}{2}Y\{\max(0, m - D_W)\}^2, \quad (1)$$

где  $D_W$  – евклидово расстояние между  $G_W(X_1)$  и  $G_W(X_2)$ ,  $Y$  – бинарная величина, равная 0, если исходные векторы лежат в одном множестве  $S_X$ , 1 иначе,  $m$  – параметр, регулирующий радиус вокруг  $G_W(X)$ .

Предложенный подход решает указанные выше проблемы, при этом обладает значимым недостатком. Для реализации такого метода необходима предобработка данных, разделение на множества  $S$ . Авторы статьи не предлагают конкретного алгоритма для такого разбиения и работают с уже размеченным датасетом. Впоследствии множества  $S_X$ ,  $\overline{S_X}$  были заменены на положительные и отрицательные пары [1], которые получаются путем аугментации входных данных.

Основной идеей контрастивного обучения является формирование пространства признаков с использованием исключительно входных данных и минимизации расстояния между

трансформированными версиями одного объекта с помощью положительных и отрицательных пар.

## 2.2 Инвариантные модели

Инвариантность – свойство, при котором объект остается неизменным под воздействием определенных преобразований. Более формально:  $f$  – инвариантна для  $G$ , если:

$$\forall x \in X, \forall g \in G \Rightarrow f(x) = f(g(x)), \quad (2)$$

где  $X, Y$  – множества,  $f : X \rightarrow Y$ ,  $G$  – множество отображений на  $X$ .

Например, модель может быть инвариантной к размытию, если размытие входных данных не изменяет выход модели.

### 2.2.1 SimCLR

В отличие от предыдущего метода в статье [3] предлагается использовать аугментацию данных вместо разделения их на множества схожих. Модель обучается минимизировать расстояние между трансформациями одного изображения и максимизировать с трансформациями другой, тем самым выявляя семантические особенности каждого входного объекта. В качестве функции потерь используется кросс-энтропия, метрики – косинусное сходство.

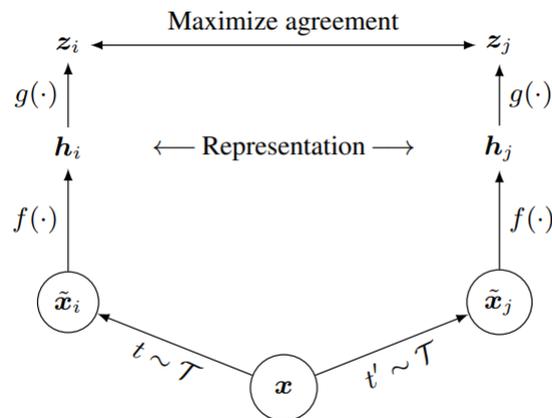


Рис. 2.1: SimCLR фреймворк.  $T$  – множество всех преобразований,  $f(\cdot)$  – энкодер,  $g(\cdot)$  – проектор,  $z_i, z_j$  – полученные эмбединги  $\tilde{x}_i, \tilde{x}_j$  соответственно.

Кроме того, используется обновленный фреймворк 2.1. При обучении используется 2 нейронные сети,  $f(\cdot)$  аналогично предыдущему методу вкладывает измененные изображения в пространство признаков  $H$ . Нелинейная  $g(\cdot)$  бьет в малоразмерное пространство вложений, к элементам которого применяется контрастная функция потерь. Для оценивания исполь-

зуется только первая часть. В статье эмперически показано, что использование проектора  $g$  позитивно сказывается на качестве модели. Также в работе представлен анализ различных трансформаций для улучшения качества модели.

Так как во время обучения, модель пытается минимизировать расстояние между конкретными трансформациями одного изображения, то она становится инвариантна к ним.

### 2.2.2 Redundancy reduction. Barlow Twins

Данный метод [11] не является контрастивным, он не использует положительные и отрицательные пары, а рассматривает корреляцию пикселей. Несмотря на это, архитектура модели аналогична SimCLR, так же есть энкодер и проектор, первому на вход подаются преобразованные изображения, второй на выходе выдает эмбединги. Однако функция потерь имеет другой вид:

$$L = \sum_i (1 - C_{ii})^2 + \lambda \sum_i \sum_{j \neq i} C_{ij}^2, \quad (3)$$

где  $C$  – матрица корреляции полученных эмбедингов  $z_k, z_l$ ,  $\lambda$  – положительный параметр. Минимизируя такую функцию потерь, модель пытается приблизить диагональные элементы к единице, а недиагональные к нулю, тем самым делая матрицу корреляции близкой к единичной. Таким образом, устремляя диагональные элементы к единице, модель обучается инвариантности, а пытаясь сделать недиагональные элементы равными нулю, модель декоррелирует компоненты разных векторов пространства вложений.

### 2.2.3 Siamese networks. SimSiam

Метод [4] использует операцию stop-gradient для обучения, в нем не задействованы отрицательные пары.

Сам фреймворк 2.2 отличается от представленных ранее. В нем так же используется энкодер  $f$ , которому на вход подаются аугментации исходных данных. При этом только к одной полученной репрезентации применяется сеть  $h$ , вктыдывающая ее в пространство эмбедингов. Затем для полученных векторов  $z_1 = h(f(x_1)), y_2 = f(x_2)$  вычисляется  $D(z_1, y_2)$  – отрицательное косинусное сходство и минимизируется среднее значение для всех изображений функции потерь:

$$L = \frac{1}{2}D(z_1, \text{stopgrad}(y_2)) + \frac{1}{2}D(z_2, \text{stopgrad}(y_1)) \quad (4)$$

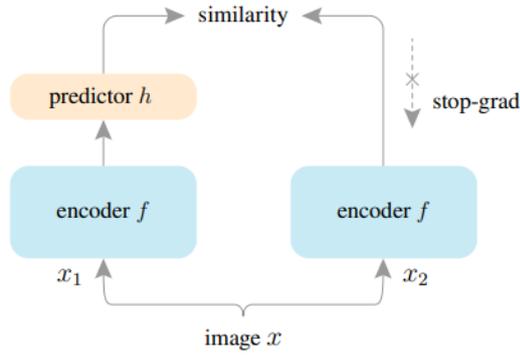


Рис. 2.2: SimSiam фреймворк.  $x_1, x_2$  – полученные после аугментации  $x$  данные,  $f(\cdot)$  – энкодер,  $h(\cdot)$  – многослойный перцептрон.

Важным отличием от вышерассмотренных подходов является stop-gradient операция (Рисунок 2.2). Благодаря ей в первом слагаемом  $y_2$  рассматривается как константа, а во втором –  $y_1$ . Таким образом, энкодер для  $x_2$  получает градиент не от  $y_2$ , а от  $z_1$  (и наоборот для  $x_1$ ). Так же в статье эмперически показана важность этой операции.

## 2.3 Эквивариантные модели

Эквивариантность – более общее свойство, частным случаем которого является инвариантность. Отображение  $f$  является эквивариантной к множеству отображений  $G$ , если:

$$\forall x \in X, \forall g \in G \exists u_g \Rightarrow f(g(x)) = u_g(f(x)), \quad (5)$$

где  $X$  – множество,  $G$  – множество отображений на  $X$ ,  $f$  – отображение из  $X$  в  $Y$ ,  $u_g : Y \rightarrow Y$  – отображение, зависящее от  $g$ .

Будем говорить, что модель эквивариантна к преобразованию  $g$ , если эмбединги входных данных полученные из нейросети будут находиться в предсказуемых местах в пространстве вложения, например, чем больше шума или размытия используем, тем больше расстояние между ними. Кроме того, в модель можно добавить дополнительный блок классификации или регрессии, для восстановления параметра преобразования  $g$ .

### 2.3.1 EquiMod

Такой подход реализован в статье [6]. В ней берется модель инвариантного обучения (SimCLR, Barlow Twins) и добавляется еще одна нейронная сеть для формирования трансформации  $u_g$  на пространстве вложений (Рисунок 2.3).

Функция потерь есть сумма инвариантной и эквивариантой части, умноженной на па-

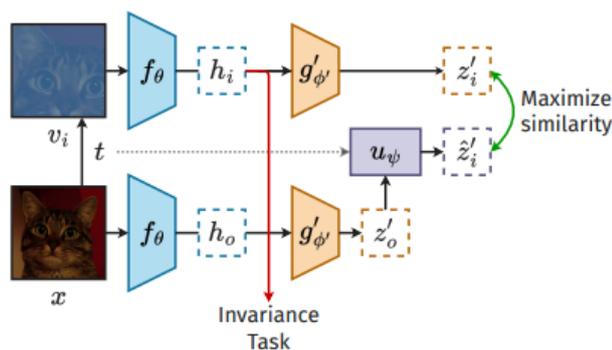


Рис. 2.3: EquiMod фреймворк.  $x$  – исходная картинка,  $v_i$  – трансформированная,  $f_\theta$  – энкодер,  $g'$  – проектор,  $u$  – предсказанная трансформация на пространстве вложений,  $\hat{z}'_i = u(z'_o)$ .

параметр  $\lambda$ . Для инвариантного слагаемого используется та же функция, что и для исходной модели, а в качестве эквивариантного авторы предлагают кросс-энтропию с температурой.

### 2.3.2 E-SSL

В статье [5] авторы обучают модель быть эквивариантной к поворотам. Для этого они также используют инвариантную модель (SimCLR, Barlow Twins), дополнительно обучая классификатор поворота изображения (Рисунок 2.4). В качестве эквивариантной функции потерь используется несмещенная оценка кросс-энтропии предсказаний и лэйблов поворота.

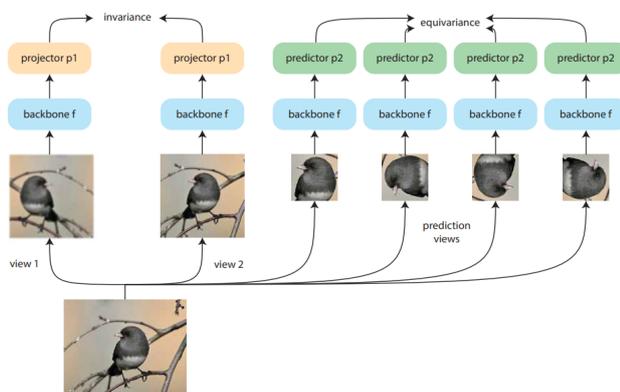


Рис. 2.4: E-SSL фреймворк. Полученная модель чувствительна к поворота и инвариантна к размытию и отображениям.

## 3 Анализ пространств признаков и эмбеддингов

Во-первых, так как описанные выше методы во время обучения отображают аугментированные данные сначала в пространство репрезентаций, затем в пространство вложений, то между этими пространствами и расположениями в них векторов должна быть зависимость.

Во-вторых, в связи с тем, что методы используют разные функции потерь, эти пространства будут обладать своей спецификой для каждого метода. Кроме того, для некоторых методов (например, Barlow Twins) не обосновано, почему обученные модели со схожими на SimCLR архитектурами на выход подают векторы признаков, а не эмбеддинги, хотя модель обучалась на последних.

### 3.1 Задачи анализа

- 1) Сравнить между собой пространства репрезентаций и эмбеддингов для моделей SimCLR, Barlow Twins, SimSiam, E-SSL как визуально, так и аналитически.
- 2) Проанализировать взаимосвязь между родственными методами и полученными ими пространствами. Например, инвариантные модели, модели, использующие одинаковые метрики или имеющие похожую архитектуру.
- 3) Проверить гипотезу для выбранных моделей: тестировать модель лучше на репрезентациях, несмотря на то, что модель обучалась на эмбеддингах.

### 3.2 Метрики, используемые при анализе

- 1) Визуальное представление пространств признаков и вложений с помощью проектирования многомерных векторов на плоскость, используя метод главных компонент (PCA). Среднее расстояние между ближайшими соседями.
- 2) Топологическая близость двух разноразмерных пространств.
- 3) Оценка внутренних размерностей пространств репрезентаций и эмбеддингов, измеренная разными методами.
- 4) Точность предсказаний тестовых данных, измеренная как на репрезентациях, так и на эмбеддингах при помощи knn.

Предложенная методика позволяет дать и визуальную, и численную характеристику пространствам для каждого метода. Метрики 1 и 2 описывают пространства геометрически и топологически, метод 3 с точки зрения теории информации, а метод 4 объединяет в себе оба подхода.

### 3.3 Архитектурные составляющие

В качестве датасета был выбран STL10, обучающая выборка – 5000 размеченных фотографий  $96 \times 96$  (500 фотографий на каждый класс), тестовая – 8000 изображений. Размер батча – 64. Количество обучающих эпох – 30 (для более ускоренного обучения). В каждом из методов энкодер – ResNet-18 без последнего слоя, в качестве проекторов используются соответствующие projection heads из библиотеки lightly. В качестве оптимизации используется SGD: lr = 0.06, momentum = 0.9. Среднее количество обучаемых параметров каждой модели равно  $11.6 \cdot 10^6$ .

Для всех тестируемых методов размерность вектора признаков равна 512, эмбеддинга – 128. При визуализации пространств используется 2000 случайных репрезентаций и эмбеддингов, соответствующих одним изображениям из тестовой выборки. В knn используется 2 и 5 соседей, были протестированы большие значения (10, 15), но значимого эффекта они не дали, еще их времязатратно использовать, поэтому было решено убрать из рассмотрения.

## 4 Экспериментальное исследование

### 4.1 Визуализация пространств

#### 4.1.1 SimCLR

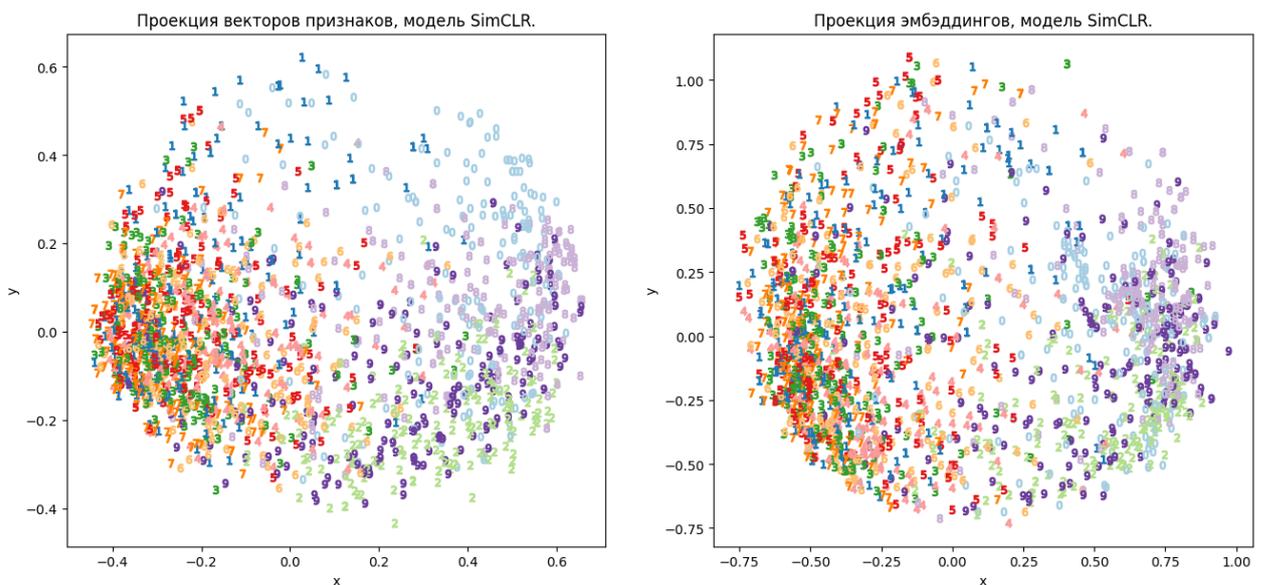
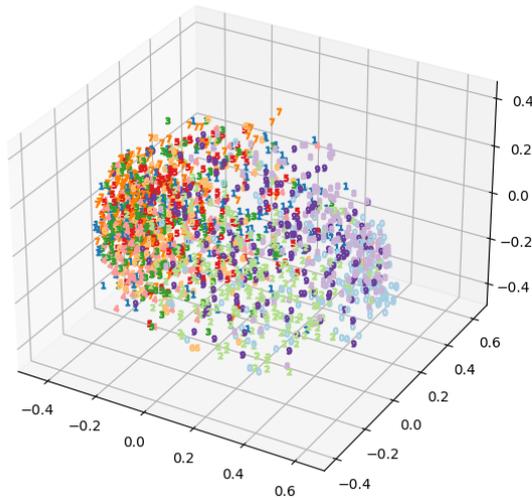


Рис. 4.1: Проекция выходов SimCLR модели на двумерное пространство, цифра + цвет – соответствующий класс объекта.

На Рисунке 4.1 на обоих графиках заметны одни и те же классы, которые удалось

Проекция репрезентаций, модель SimCLR.



Проекция эмбеддингов, модель SimCLR.

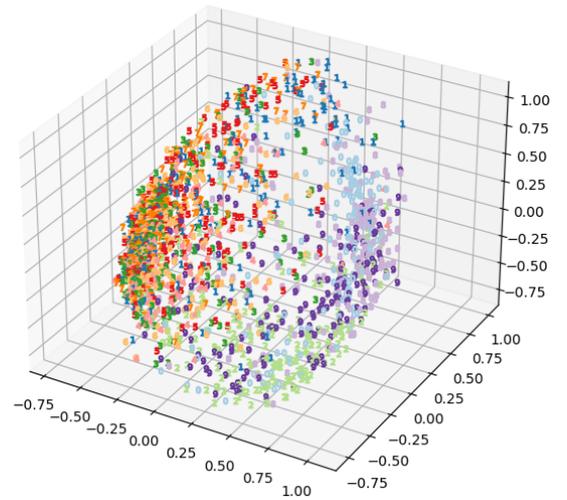
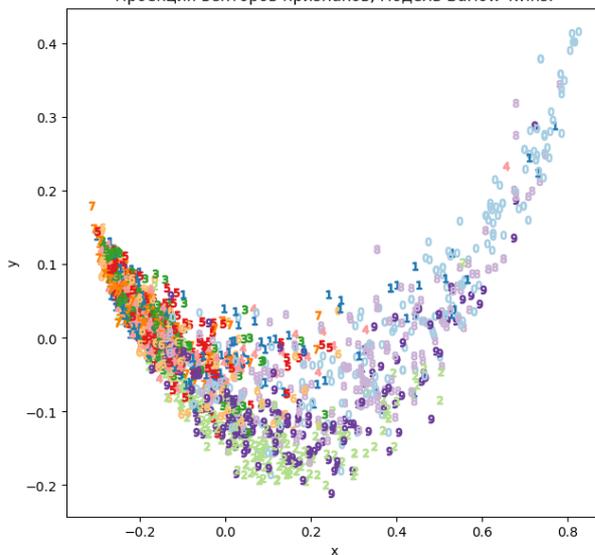


Рис. 4.2: Проекция выходов SimCLR модели на трехмерное пространство, цифра + цвет – соответствующий класс объекта.

отделить модели. При этом отметим, что эмбеддинги располагаются более равномерно по окружности, в отличие от репрезентаций. Модель справилась с отделением транспорта (0, 2, 8, 9) от животных (остальные индексы). На трехмерной проекции (Рисунок 4.2) заметно, что эмбеддинги в среднем расположены дальше друг от друга, чем вектора репрезентаций. Действительно, среднее расстояние между соседними репрезентациями = 0.189, а между эмбеддингами = 0.230.

#### 4.1.2 Barlow Twins

Проекция векторов признаков, модель Barlow Twins.



Проекция эмбеддингов, модель Barlow Twins.

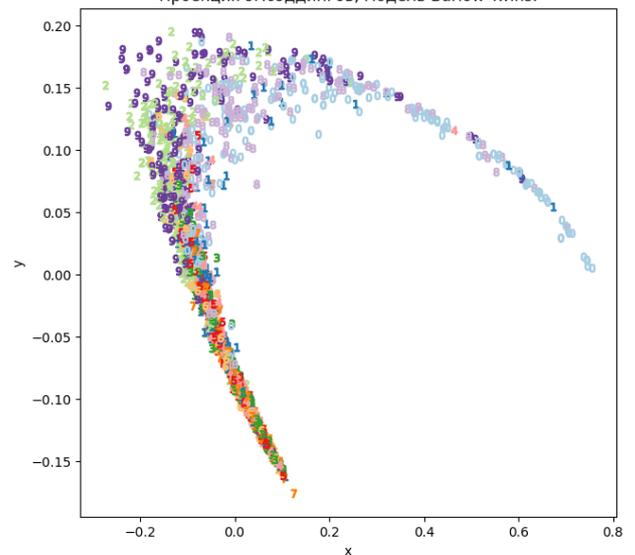


Рис. 4.3: Проекция выходов модели Barlow Twins на двумерное пространство, цифра + цвет – соответствующий класс объекта.

В отличие от предыдущего метода, проекции и репрезентаций, и эмбеддингов образуют параболу. Также видно, что эмбеддинги по сравнению с репрезентациями располагаются более плотно друг к другу. Численное подтверждение: среднее расстояние между ближайшими соседями в пространстве эмбеддингов в  $0.034/0.005 \approx 6.8$  раз меньше, чем в пространстве векторов признаков.

### 4.1.3 SimSiam

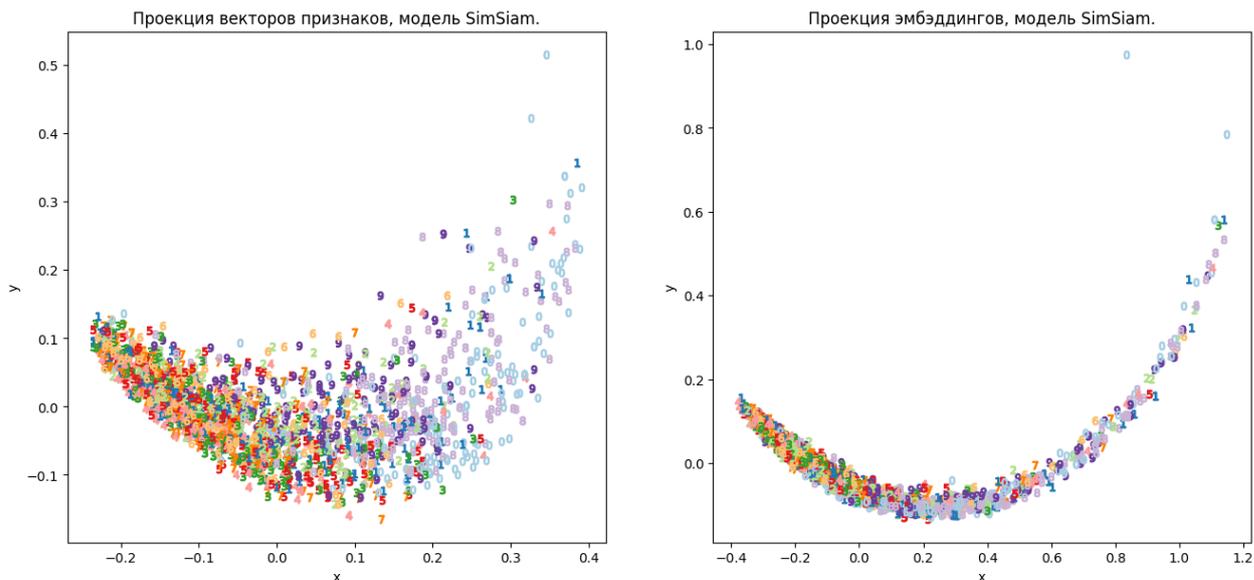


Рис. 4.4: Проекция выходов модели SimSiam на двумерное пространство, цифра + цвет – соответствующий класс объекта.

Аналогично модели Barlow Twins проекции образуют дугу параболы, при этом снова эмбеддинги располагаются более плотно. Среднее расстояние между соседними репрезентациями равно 0.143, а между соседними эмбеддингами – 0.054. При этом среднее расстояние между объектами одного класса примерно на таком же уровне (Таблица 5.1).

### 4.1.4 E-SSL

Так как модель обучалась, используя SimCLR подход, то полученные пространства (Рисунок 4.5) сравнимы с Рисунок 4.1: так же проецируются на окружность, при этом значительные отличия от SimCLR заметны при проектировании на 3D плоскость (Рисунок 5.3). Кроме того, отличаются и численные характеристики расстояний. В то время, как в SimCLR среднее расстояние между эмбеддингами больше чем между репрезентациями, при E-SSL методе наоборот. Эмбеддинги – 0.097, векторы признаков – 0.230.

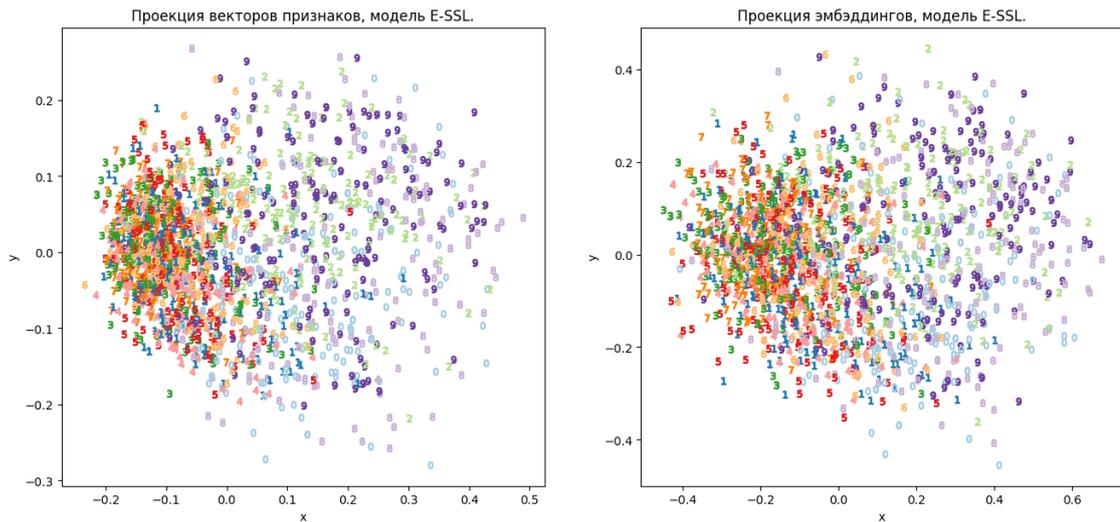


Рис. 4.5: Проекция выходов модели E-SSL на двумерное пространство, цифра + цвет – соответствующий класс объекта. Модель обучалась эквивариантности к поворотам на  $90^\circ$ , в качестве инвариантной основы был взят SimCLR.

## 4.2 Геометрическое и топологическое сравнение

### 4.2.1 Оценка внутренней размерности

С помощью оценки внутренней размерности можно описать, насколько сложно или полно представлены данные в пространстве. Эта оценка должна учитываться при подборе количества скрытых слоев нейронной сети или формировании ее функции. Для оценивания внутренней размерности использовались 2 метода [7, 9]. Результаты представлены в Таблице 4.1

Таблица 4.1: Оценки внутренних размерностей репрезентаций и эмбеддингов для каждой модели, репрезентации имеют размерность 512, эмбеддинги – 128.

	PSA-based		MLE	
	Repr	Emb	Repr	Emb
SimCLR	26	9	12.8	6.7
Barlow Twins	4	2	8.9	3.5
SimSiam	70	3	23.9	9.1
E-SSL	250	7	29.9	8.4
<b>Среднее</b>	<b>87.5</b>	<b>5.25</b>	<b>18.875</b>	<b>6.925</b>

Наибольшая внутренняя размерность репрезентаций у E-SSL – эквивариантного метода, это и предсказуемо, так как кроме информации о самой картинке, модели необходимо запоминать примененные трансформации – отсюда и увеличение внутренней размерности.

## 4.2.2 Топологическая оценка сходства пространств

Для сравнения пространств репрезентаций и эмбэдингов был использован метод расхождений топологии представлений (RTD, статья [2]). Данный метод сравнивает каждую репрезентацию с ее соответствующим эмбэдингом, тем самым пытается оценить, насколько хорошо одно пространство приближается другим. Аналогично строится оценка и в другую сторону. Таким образом, чем ближе среднее этих оценок к нулю, тем более похожи топологические представления. Результаты оценок представлены в Таблице 4.2.

Таблица 4.2: Похожесть структуры пространства признаков и пространства вложений. Измерено с помощью метода RTD.

	Repr	Emb	Среднее
SimCLR	–	0.143	–
Barlow Twins	0.038	0.016	0.027
SimSiam	0.07	0.053	0.062
E-SSL	0.074	0.143	0.109

Столбец **Repr** показывает, насколько хорошо пространство признаков приближается пространством вложений в топологическом смысле. Столбец **Emb** – наоборот. Видно, что для каждого из методов значения метрик близки к нулю. Значит, несмотря на наличие нелинейного проектора в каждой из модели топология пространства остается почти неизменной.

## 4.3 Точность предсказаний, knn

Для подсчета точности использовался метод  $k$ -ближайших соседей. Тестирование проводилось как на репрезентациях, так и на эмбэдингах.

Таблица 4.3: Точность предсказаний в процентах, измеренная на репрезентациях и эмбэдингах с помощью  $k$  ближайших соседей, сам объект не учитывался.

	$k = 2$		$k = 5$	
	Repr	Emb	Repr	Emb
SimCLR	39.3	34.35	<b>37.975</b>	33.45
Barlow Twins	28.925	23.313	26.444	<b>22.237</b>
SimSiam	36.363	27.4	32.916	24.972
E-SSL	<b>41.088</b>	26.538	36.568	24.762
<b>Среднее</b>	<b>36.419</b>	<b>27.9</b>	<b>33.476</b>	<b>26.355</b>

Результаты в Таблице 4.3 подтверждают необходимость использования репрезентаций, а не эмбэдингов в качестве выходов модели для каждого из подходов. Заметим также, что методы SimSiam и E-SSL имеют наибольшую разность между точностью для векто-

ров признаков и эмбэдингов. Возможно, это обусловлено наличием дополнительной головы-предиктора в моделях, тем самым обучение релаксируется и проектор недообучается.

## 5 Выводы

Основные цели проекта достигнуты: изучены основные концепции обучения без учителя, реализованы несколько из существующих подходов, проанализированы представления данных внутри моделей, эмперически показано, что в качестве выходных данных обученной модели необходимо брать репрезентации.

### 5.1 Результаты

- 1) У эквивариантных моделей пространство репрезентаций более насыщено, чем у инвариантных.
- 2) Геометрически и топологически пространства репрезентаций и вложений имеют практически идентичную структуру, несмотря на значительную разницу в размерностях. Отличия заключаются в расстояниях между векторами.
- 3) Сформулирована гипотеза, что чем больше в модели дополнительных сетей, заточенных под конкретные задания (например, предсказатель для трансформации в E-SSL), то тем выше разность точности предсказаний на репрезентациях и на эмбэдингах.

### 5.2 Перспективы дальнейшей работы

При дальнейших исследованиях можно расширить выборку рассматриваемых методов, протестировать на них сформулированную выше гипотезу. Также можно использовать более сложные методы топологического анализа для выявления закономерностей в пространствах. Можно рассмотреть, как меняются эти пространства при воздействии конкретных трансформаций при аугментации.

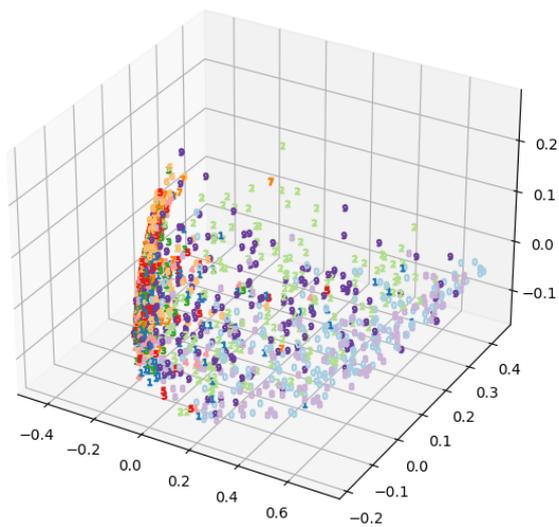
## Список литературы

- [1] Randall Balestriero, Mark Ibrahim и Vlad Sobal. “A Cookbook of Self-Supervised Learning”. В: *arXiv preprint arXiv:2304.12210* (2023).
- [2] Serguei Barannikov, Ilya Trofimov, Nikita Balabin и Evgeny Burnaev. “Representation Topology Divergence: a Method for Comparing Neural Network Representations”. В: *arXiv preprint arXiv:2201.00058* (2022).
- [3] Ting Chen, Simon Kornblith, Mohammad Norouzi и Geoffrey Hinton. “A Simple Framework for Contrastive Learning of Visual Representations”. В: *arXiv preprint arXiv:2002.05709* (2020).
- [4] Xinlei Chen и Kaiming He. “Exploring Simple Siamese Representation Learning”. В: *arXiv preprint arXiv:2011.10566* (2020).
- [5] Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung и Marin Soljačić Pulkit Agrawal. “Equivariant Contrastive Learning”. В: *arXiv preprint arXiv:2111.00899v2* (2022).
- [6] Alexandre Devillers и Mathieu Lefort. “EquiMod: An Equivariance Module to Improve Self-Supervised Learning”. В: *arXiv preprint arXiv:2211.01244* (2022).
- [7] Mingyu Fan, Nannan Gu, Hong Qiao и Bo Zhang. “Intrinsic dimension estimation of data by principal component analysis”. В: *arXiv preprint arXiv:1002.2050* (2010).
- [8] Raia Hadsell, Sumit Chopra и Yann LeCun. “Dimensionality Reduction by Learning an Invariant Mapping”. В: *Computer Society Conference on Computer Vision and Pattern Recognition* (2005).
- [9] Elizaveta Levina и Peter J. Bickel. “Maximum Likelihood Estimation of Intrinsic Dimension”. В: *Advances in Neural Information Processing Systems* (2005).
- [10] Tete Xiao, Xiaolong Wang, Alexei A. Efros и Trevor Darrell. “What should not be contrastive in contrastive learning”. В: *arXiv preprint arXiv:2008.05659* (2020).
- [11] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun и Stéphane Deny. “Barlow Twins: Self-Supervised Learning via Redundancy Reduction”. В: *arXiv preprint arXiv:2103.03230* (2021).

# Приложения

## Приложение А

Проекция репрезентаций, модель Barlow Twins.



Проекция эмбэдингов, модель Barlow Twins.

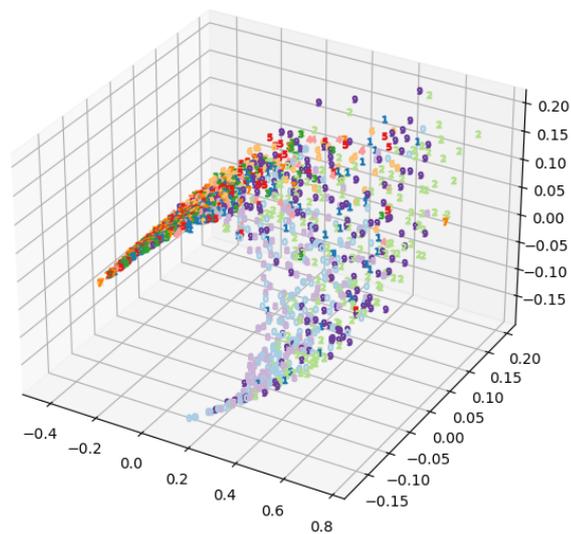
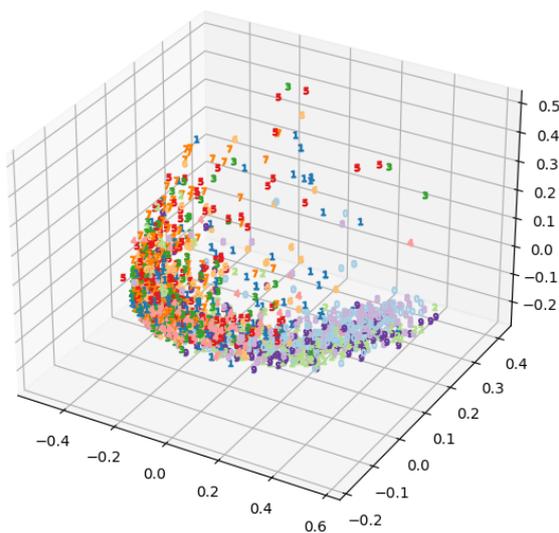


Рис. 5.1: Проекция выходов модели Barlow Twins на трехмерное пространство.

Проекция репрезентаций, модель SimSiam.



Проекция эмбэдингов, модель SimSiam.

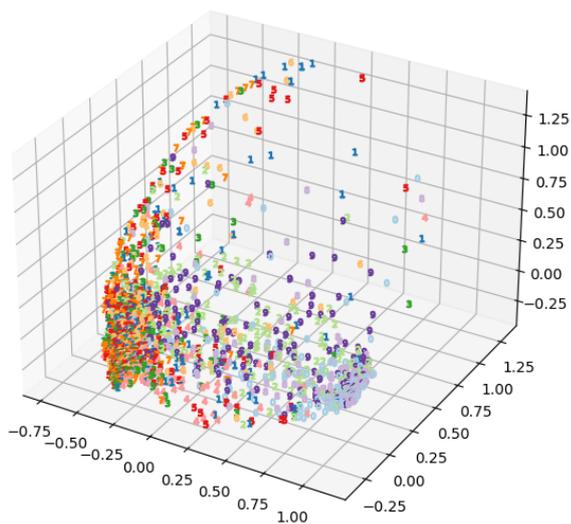
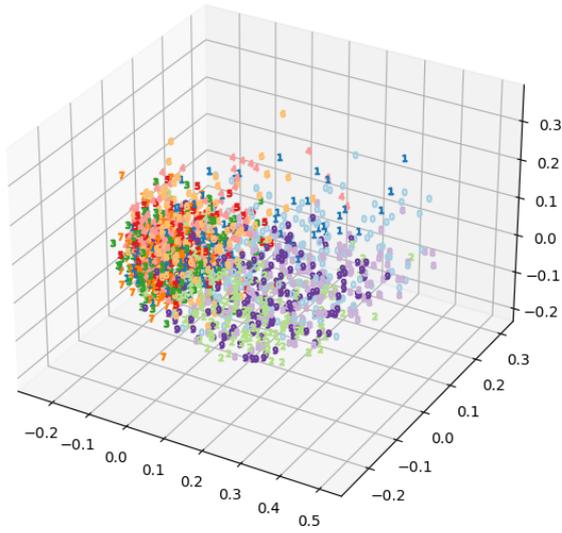


Рис. 5.2: Проекция выходов модели SimSiam на трехмерное пространство.

Проекция репрезентаций, модель E-SSL.



Проекция эмбеддингов, модель E-SSL.

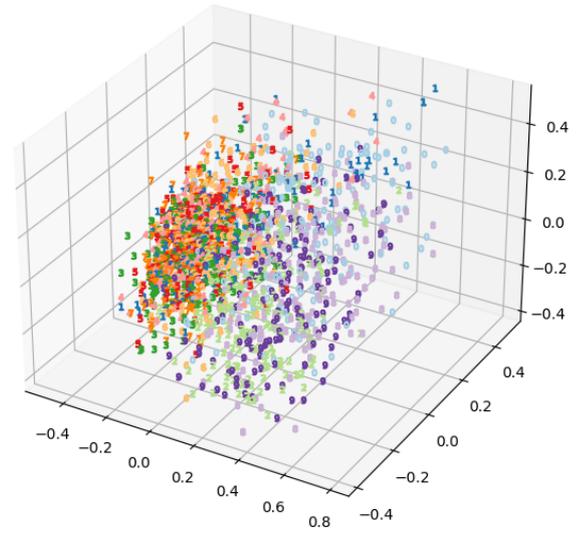


Рис. 5.3: Проекция выходов модели E-SSL на трехмерное пространство.

## Приложение Б

Таблица 5.1: Среднее расстояние до ближайшего соседа того же класса, сам элемент не учитывается.

	Repr	Emb
SimCLR	0.18	0.21
Barlow Twins	0.035	0.005
SimSiam	0.134	0.05
E-SSL	0.229	0.1