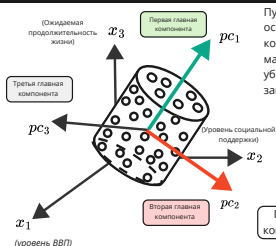


## Занятие 6

### Обучение без учителя: снижение размерности

#### PCA



Пусть данные имеют форму цилиндра с эллипсом в основании. Метод PCA подбирает определённую систему координат. Оси этой системы соответствуют направлениям максимальной дисперсии в данных и нумеруются в порядке убывания дисперсии. Координаты элементов в ней записываются через найденные коэффициенты  $a_{ij}$  так:

$$\begin{aligned} pc_1 &= a_{11} \cdot x_1 + a_{12} \cdot x_2 + a_{13} \cdot x_3 \\ pc_2 &= a_{21} \cdot x_1 + a_{22} \cdot x_2 + a_{23} \cdot x_3 \\ pc_3 &= a_{31} \cdot x_1 + a_{32} \cdot x_2 + a_{33} \cdot x_3 \end{aligned}$$

Главные компоненты      Веса      Изначальные признаки

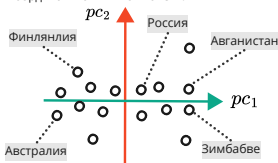
Степени влияния исходных признаков на каждую главную компоненту.

Очевидно, вдоль направления оси цилиндра дисперсия данных максимальна (Проводим  $pc_1$ )  
Вдоль большой оси основания цилиндра дисперсия меньше (Проводим  $pc_2$ )  
Вдоль малой оси основания цилиндра дисперсия ещё меньше (Проводим  $pc_3$ )

#### Визуализация результатов

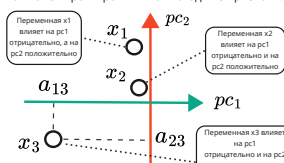
##### График наблюдений

Позволяет визуализировать данные в системе координат главных компонент.



##### График коэффициентов

Позволяет определить, как на значения компонент  $pc_1$  и  $pc_2$  влияют исходные признаки.



#### Оценка качества снижения размерности и выбор оптимального числа главных компонент.

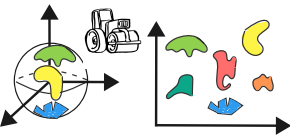
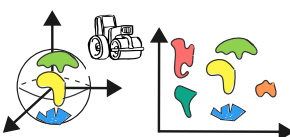



Задача нахождения направления максимальной дисперсии решается методом множителей Лагранжа. В её результате находятся собственные значения, и соответствующие им собственные векторы для ковариационной матрицы ( $X^T \cdot X$ ), где  $X$  - матрица ( $n\_samples \times n\_features$ ). Собственными векторами являются главные компоненты, а соответствующие им собственные значения являются объяснёнными дисперсиями.

	Объяснённая дисперсия	Доля объяснённой дисперсии	99.98% 100%
PCA 1	46.73	99.25 %	
PCA 2	0.35	0.73 %	
PCA 3	0.01	0.02 %	



## Занятие 6

### Обучение без учителя: снижение размерности

Продвинутые методы понижения размерности	
TSNE	UMAP
<p>Алгоритм SNE представлен Джеффри Хинтоном в 2002 году. tSNE (t-distributed stochastic neighbor embedding) опубликован Лоренс ван дер Маатен в соавторстве с Джеффри Хинтоном в 2008 году.</p> 	<p>Это самый свежий алгоритм, он появился в 2018 году - UMAP (Uniform Manifold Approximation and Projection). Принцип работы аналогичен t-SNE. Посмотрим и на его результаты.</p> 
<p><b>Основная идея:</b> скатие до пространства меньшей размерности с сохранением расстояния между близкими точками. При этом могут использоваться искажения двух видов: сжатия/растяжения многомерной структуры, а также разрывы в разных местах. Точки, которые были рядом оказываются разнесены друг от друга.</p> <p><b>Параметры для настройки:</b></p> <ol style="list-style-type: none"> <li><b>initial_dims (default=50).</b> Является количеством главных компонент, оставленных после первоначального снижения размерности. tSNE плохо работает с большим количеством измерений, поэтому перед запуском алгоритма, изначальная размерность понижается до initial dims при помощи PCA. Значение по умолчанию часто оказывается оптимальным. Для более тонкой настройки строится график объяснённой дисперсии и выбирается оптимальное количество главных компонент.</li> <li><b>perplexity (default=30).</b> Является предположением о количестве близких соседей, которые есть у каждой точки. Типичные значения 5-50. Чем больше perplexity, тем больше учитывается глобальная структура и меньше локальная. Параметр не может быть больше, чем количество точек. При настройке стоит ориентироваться на формулу <math>Perplexity_{ideal} \approx \sqrt{N}</math> [см. материал 1 внизу]</li> <li><b>max_iter (default=100).</b> Алгоритмы обучаются методом градиентного спуска (минимизируется специальная функция потерь [см. материал 3 ниже]). Параметр показывает количество итераций.</li> </ol>	
Сравнительный анализ	
<ol style="list-style-type: none"> <li><b>TSNE вычислительно сложнее.</b> В TSNE снижение размерности до более чем 2 компонент проблематично из-за сложности вычислений! В то же время UMAP быстро справляется с созданием, например, 30 UMAP-компонент.</li> <li><b>UMAP лучше сохраняет глобальную структуру в данных.</b> Визуализация работы алгоритмов выше является преувеличением, но в то же время хорошо иллюстрирует этого пункта. TSNE может демонстрировать перемешивание кластеров в то время как UMAP лучше сохраняет их взаимное расположение. [см. материал 3 ниже]</li> <li><b>TSNE чувствительнее к параметру perplexity.</b> В случае использования этого алгоритма, стоит серьёзно подойти к настройке этого параметра. [см. материал 1 ниже]</li> <li><b>Кластеризацию (и методы обучения с учителем) стоит проводить на десятках-сотнях компонентах UMAP или PCA.</b> Обучение на оригинальном датасете, содержащем множество измерений (тысячи и десятки тысяч), не является хорошей идеей из-за проклятия размерности (многомерные векторы являются малоинформативными; в многомерном пространстве все объекты начинают сильно отличаться друг от друга [см. <a href="#">rarity Machine Learning article</a> Curse of dimensionality in Wikipedia]). В то же время обучение на 2 компонентах tSNE является слишком экстремальной.</li> <li><b>UMAP позволяет добавлять новые данные (делать transform() на новых данных), в то время как TSNE лишён такой возможности.</b> Это основная причина, по которой метод TSNE используют только для визуализации (в крайнем случае для кластеризации), в то время как метод UMAP широко распространён в том числе в моделях обучения с учителем.</li> </ol>	
<p>Материал 1. <a href="#">Статья на Medium</a> про подбор гиперпараметров для TSNE.      Материал 2. <a href="#">Визуализация работы алгоритма, самоучитель по настройке его параметров и интерпретации результатов.</a>      Материал 3. <a href="#">Статья на Medium</a> про сравнение методов tSNE и UMAP.</p>	
  	

Занятие 6  
Обучение без учителя: снижение размерности

Кластеризация	
Основная идея	
Заранее сообщаем алгоритму количество	Алгоритм сам определяет количество кластеров
Метрики кластеризации: silhouette, AMI, Davies-Bouldin, Silhouette, Silhouette Coefficient	
Kmeans, Spectral Clustering	DBSCAN