T-SNE (t-Distributed Stochastic Neighbor Embedding) — метод визуализации многомерных данных, который позволяет представлять высокоразмерные данные в виде графических изображений в двумерном или трёхмерном пространстве. Метод t-SNE даёт возможность сохранять структуру данных в процессе снижения размерности, сохраняя при этом связи между точками.

Первый этап: расчёт условных вероятностей сходства между точками данных

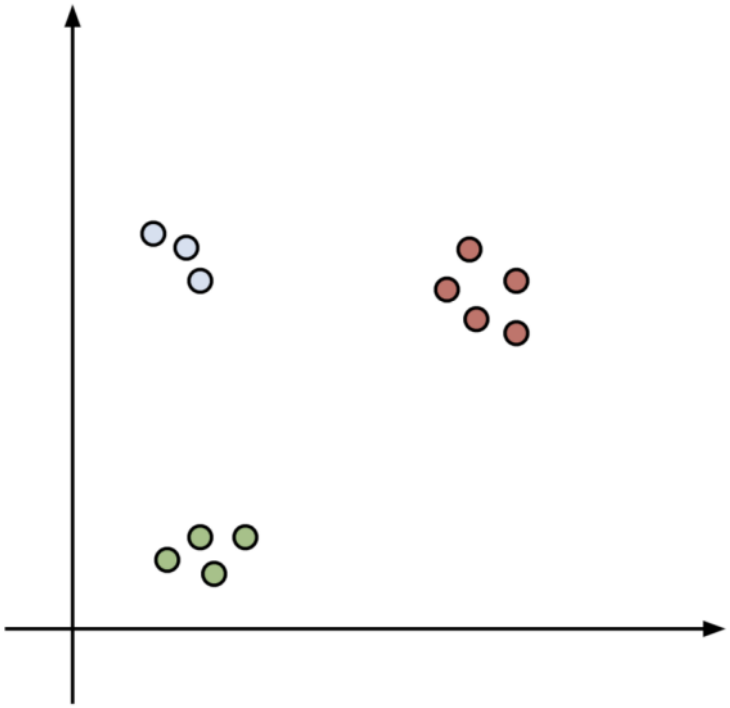
Первый этап алгоритма t-SNE — вычисление евклидовых расстояний между каждой точкой и всеми остальными точками. Используя эти расстояния, мы преобразуем их в условные вероятности, которые представляют собой меру сходства между каждыми двумя точками. Это означает, что мы хотим оценить, насколько похожи между собой две точки в данных, или, другими словами, насколько вероятно, что они являются соседними.

Условная вероятность того, что точка x_j находится рядом с точкой x_i, представлена **гауссовским распределением** с центром в точке x_i и стандартным отклонением σ_i. Математически это записывается следующим образом:

$$p_{j|i} = \frac{exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \quad exp(-||x_i - x_k||^2/2\sigma_i^2)}$$

Вероятность того, что точки x_i и x_j располагаются по соседству

Причина деления на сумму всех других точек, размещённых в гауссовском распределении с центром в точке x_i, заключается в том, что мы можем столкнуться с кластерами различной плотности. Чтобы объяснить это, обратимся к примеру из видео:



Точки в высокоразмерном пространстве

Изображение: Вячеслав Чарин

Как видите, плотность коричневого кластера ниже, чем плотность зелёного. Поэтому, если мы вычисляем сходства между каждыми двумя точками только по гауссовскому распределению, увидим меньше сходства между точками коричневого цвета по сравнению с точками зелёного цвета. В конечном счёте мы не будем обращать внимания на то, что у некоторых кластеров разная плотность, мы просто хотим увидеть их как кластеры и поэтому выполняем эту нормализацию.

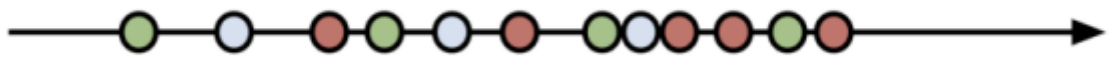
Используя созданные условные распределения, вычисляем совместное распределение вероятностей с помощью следующего выражения:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

Второй этап: создание набора точек в пространстве меньшей размерности

На этом этапе мы создаём набор представленных точек в пространстве меньшей размерности, а также рассчитываем для них совместное распределение вероятностей.

Для этого строим случайный набор точек (с тем же количеством точек, что и в исходном наборе данных), в котором будет K признаков, где K — наша целевая размерность. Обычно, если мы хотим снизить размерность для визуализации, K будет равно 2 или 3. Если вернуться к примеру, на этом этапе алгоритм строит случайный набор точек в одном измерении — на прямой:



Случайный набор точек в одном измерении

Изображение: Вячеслав Чарин

Для этого набора точек мы создадим их совместное распределение вероятностей, но на этот раз будем использовать **распределение Стьюдента (t-распределение)**, а не гауссовское распределение, как для исходного набора данных (t в t-SNE как раз означает t-распределение). Здесь будем обозначать вероятности как q, а точки — как y.

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq l} \quad (1 + ||y_k - y_l||^2)^{-1}}$$

Причина выбора t-распределения вместо гауссовского распределения — его свойство тяжёлых хвостов. Это свойство приводит к тому, что умеренные расстояния между точками в пространстве высокой размерности становятся экстремальными в пространстве меньшей размерности, что помогает предотвратить скученность точек в меньшей размерности. Ещё одно

преимущество t-распределения — улучшение процесса оптимизации на третьем этапе алгоритма.

Третий этап: оптимизация KL-дивергенции

Теперь мы используем **расхождение Кульбака — Лейблера (KL-дивергенцию)**, чтобы сделать совместное распределение вероятностей точек данных в пространстве меньшей размерности максимально похожим на распределение из исходного набора данных. Если процесс преобразования будет успешным, мы получим хорошее сокращение размерности.

KL-дивергенция — мера того, насколько два распределения различаются между собой. Для распределений P и Q в пространстве вероятностей X KL-расхождение определяется следующим образом:

$$D_{KL}(P||Q) = \sum_{x \epsilon X} P(x) log \frac{P(x)}{Q(x)}$$

Определение KL-дивергенции между распределениями вероятностей P и Q

Чем более похожи распределения между собой, тем меньше значение KL-расхождения (достигает нуля, когда распределения идентичны).

Возвращаемся к нашему алгоритму: мы пытаемся изменить набор данных в пространстве меньшей размерности таким образом, чтобы его совместное распределение вероятностей было максимально похожим на распределение из исходного набора данных. Это делается с помощью **градиентного спуска**. Функция ошибки, которую градиентный спуск пытается минимизировать, — это KL-дивергенция между совместным распределением вероятностей P из пространства высокой размерности и Q — из пространства меньшей размерности.

$$C = KL(P||Q) = \sum_i \sum_j p_{ij} log \frac{p_{ij}}{q_{ij}}$$

Функция ошибки для градиентного спуска

Из этой оптимизации получаем значения точек в наборе данных меньшей размерности и используем их для визуализации. В нашем примере мы видим кластеры в пространстве меньшей размерности следующим образом:

Изображение: Вячеслав Чарин