План—Конспект по курсу Компьютерное зрение

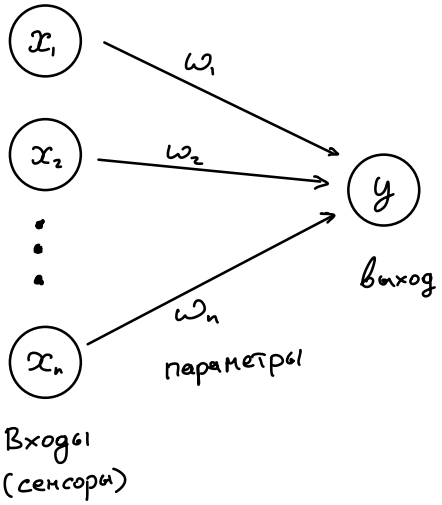
Постановка задачи распознавания изображений на естественном языке.

1. нет математической модели
2. нет математически строгой формулировки цели
3. нет точного описания входных данных

Непонятно, что называть решением: например, при повороте цифры "6", когда шестерка перестанет быть шестеркой, и когда она станет девяткой?

Идея: человек распознает изображения, значит достаточно хорошая модель человеческого мозга тоже сможет распознавать картинки.

Персептрон Розенблатта — простая модель человеческого нейрона.



$$y \overset{def}{=} \text{Sign}\left(\sum_k w_k x_k\right)$$

На вход персептрону подается изображение (в каком-то виде), а выход интерпретируется как результат распознавания.

Для распознавания двух классов (например, "кот" или "не кот") одному классу соответсвует выход "+1", а другому "-1".

Вектор «**w**» является параметром, который требуется найти так, чтобы распознавание было верным.

Каким образом искать «**w**»?
Какие значения будут "лучшими"?

Идея: поставить задачу поиска «**w**», как задачу минимизации функционала суммарной ошибки по всем входным картинкам.

$$Q(w) = \sum_x \left(y(x,w) - \hat{y}(x)\right)^2, \quad w = \arg\min Q(w)$$

$y(x,w)$ — выход персептрона; $\hat{y}(x)$ — правильный класс, т.е. выход идеального классификатора

А что такое "все входные картинки"?
Черный квадрат является допустимым входом?

Идея: если мы построим классификатор, который будет хорошо работать
на заранее выбранном *репрезентативном* множестве входных данных,
то он будет работать и на всех других данных.

Размеченные данные делим на:
1. тренировочный набор данных
2. тестовый набор данных

Как искать **argmin J(w)** ?

Градиентный спуск? Но **sign(.)** не дифференцируемая функция!

Идея: заменить **sign** на похожую, но дифференцируемую функцию,
т.е. заменить на сигмоиду.

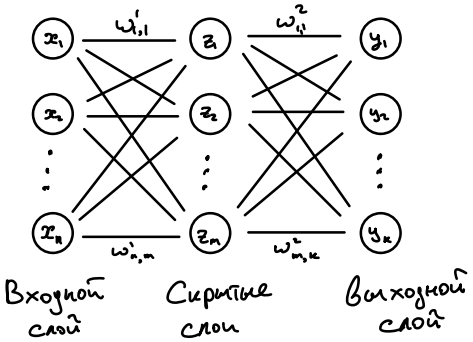
$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Может ли персептрон решать задачи распознавания изображений?

Персептрон строит разделяющую гиперплоскость в пространстве признаков,
т.е. для успешного решения задачи нужно, чтобы классы были линейно разделимы.

Идея: объединить много персептронов в одну большую сеть.

Deep Neural Network



Каждые два последовательных слоя образуют полный двудольный граф.

Что может дать такая искусственная нейронная сеть?

Любую булеву функцию можно реализовать в виде нейронной сети,
т.е. ограничивая себя такой формой классификатора, мы себя ничем
не ограничиваем, но и соответственно, задачу не упрощаем.

Как интерпретировать выход сети если мы хотим одновременно распознавать не два, а много классов?

Идея: интерпретировать выход сети (вектор значений нейронов последнего слоя),
как функцию распределения вероятности принадлежности соответствующему классу.

Требуется нормировка выхода: **softmax**

$$P_k = \frac{e^{y_k}}{\sum_k e^{y_k}}$$

Вопрос: а где тут вероятностное пространство? Где случайные события?

Как построить целевой функционал?

Идея: минимизировать расстояние Кульбака—Лейблера до идеального распределения, т.е. до распределения у которого вероятность 1 на нужном классе и 0 на всех остальных.

$D(p,q) = H(p|q) - H(p)$

Интерпретация энтропии как меры информации, коды Хаффмана.
Кросс-энтропия.

Что делать если в логарифме стоит 0?

Идея: заменить 0 на маленькое число, например на 10^{-10}.

Чем плоха сигмоида?

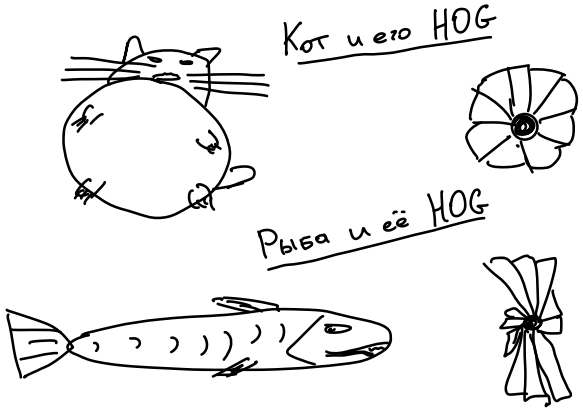
Идея: заменить сигмоиду на $ReLu(x) = max(0, x)$

Проблемы применения построенного классификатора для распознавания изображений:
1. никак не учитываем соседство пикселей
2. никак не учитываем взаимное расположение пикселей на некотором расстоянии друг от друга

Нужны механизмы выделения признаков из изображений, которые бы учитывали взаимное расположение пикселей, тогда вектор этих признаков можно было бы подать на построенный классификатор.

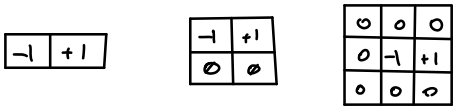
Какие признаки можно придумать для изображений?

HOG Гистограмма ориентированных градиентов



Идея: внутри одного класса гистограммы ориентированных градиентов похожи, а между классами они сильно различаются, значит эту гистограмму разумно взять в качестве вектора признаков.

Идея: вычислять градиенты с помощью операции свертки с фильтром.



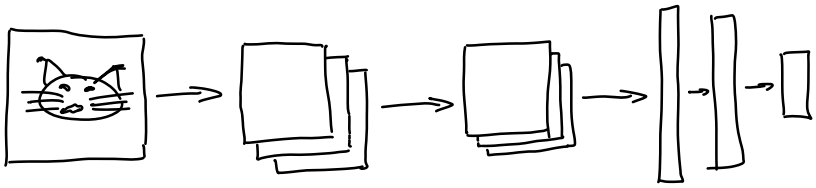
Примеры фильтров:
1. размытие Гаусса
2. резкость
3. выделение границ (Превитта, Собеля)

Как находить хорошие фильтры для выделения признаков?

Идея: пусть поиск фильтров будет частью обучения классификатора, т.е. значения в матрице фильтра это настраиваемые параметры.

w_{11}	w_{12}	w_{13}
w_{21}	w_{22}	w_{23}
w_{31}	w_{32}	w_{33}

Структура Сверточной нейронной сети (CNN)

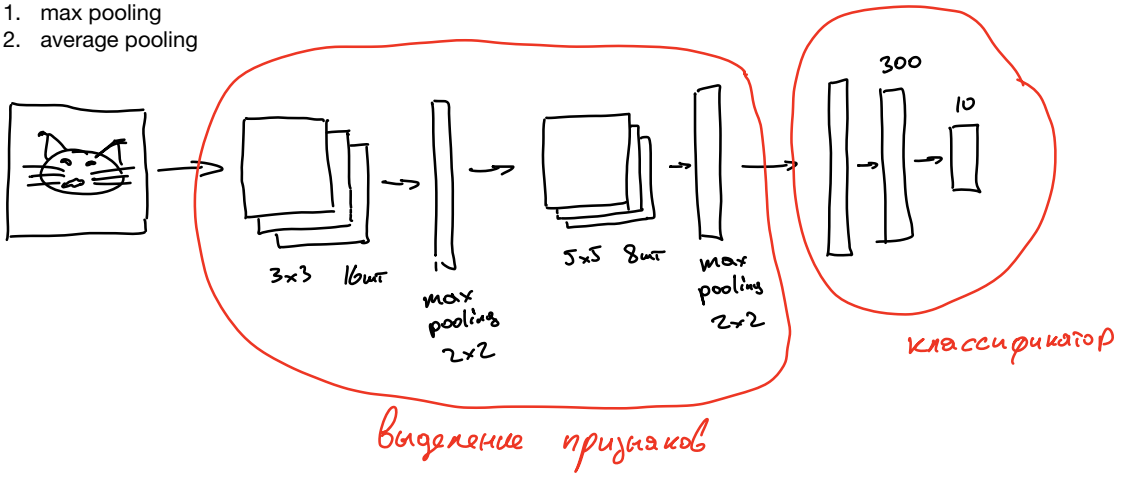


Идея: использовать несколько последовательных слоев фильтров, тем самым создавая признаки из признаков, т.е. признаки более высокого уровня; таким образом мы учитываем взаимное расположение пикселей.

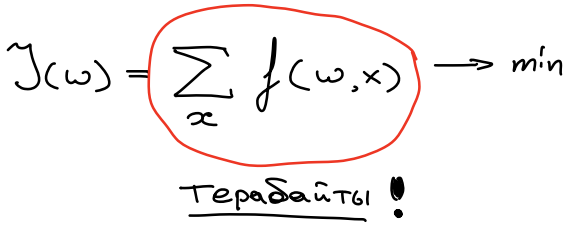
Проблема: получается ОЧЕНЬ много параметров!

Идея: соседние значения в матрице признаков содержат близкую информацию, значит можно их заменить на одно значение, тем самым существенно уменьшить размерность.

Pooling:
1. max pooling
2. average pooling

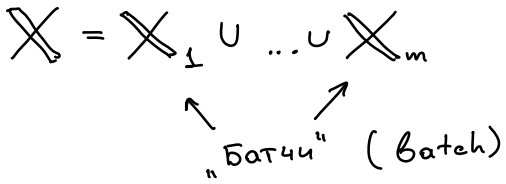


Проблема: вся сумма функционала не влезает в память компьютера!



Идея: рассчитывать шаги градиентного спуска по частичным суммам.

Мотивация: метод Роббинса—Монро.



Алгоритм:

$$\omega_0 = \text{Random}$$

$$\omega_{k+1} = \omega_k - \lambda_k \cdot \nabla_\omega \sum_{x \in X_k} f(x, \omega), \quad \lambda_k = \frac{1}{k}$$

Эпоха это пробег по всем батчам.
Для обучения нужно пройти несколько эпох.

Будет ли алгоритм сходиться?

Как понять, что началось переобучение?

Пример хорошей сети: AlexNet (2015 год).

Практическое задание:

1. построить и обучить сеть для распознавания рукописных цифр, MNIST
2. найти момент, когда сеть начала переобучаться
3. посмотреть с каким качеством обученая сеть распознает ваш почерк
4. постараться построить сеть с минимальным количеством параметров и качеством распознавание не менее 0.98