

# Машинное обучение: задание 1

Составитель: Виктор Кантор

20 февраля 2017 г.

## Организационные вопросы

### Дедлайн

На это задание дается одна неделя, т.е. после выдачи задания прислать его нужно до следующего семинара в вашей группе. Задание выдается на неделю с 20 февраля, дедлайн выпадает на неделю с 27 февраля.

Пример: если вам выдали задание на семинаре в понедельник 20 февраля, то задание нужно сделать и отослать до начала семинара в понедельник 27 февраля.

### Сдача задания

1. Заведите свой репозиторий на github для домашних заданий по курсу
2. Выполненные задания сохраните в ipynb (в случае практических задач) и pdf (в случае теоретических) и сделайте push в свой репозиторий. Желательно для каждого домашнего задания в дальнейшем заводить отдельную папку.
3. Пришлите на почту ml.course.mipt@gmail.com ссылку на выложенное на github решение задания. На всякий случай продублируйте в письме ipynb и pdf файлы. Тему письма укажите в формате «ML2017\_fall <номер\_группы> <фамилия> <имя>, Задание 1 (Introduction)», например: «ML2017\_fall 491 Страуструп Бьярн, Задание 1 (Introduction)»
4. Если есть принципиальное желание оформлять теоретические задачи в ipynb в Markdown-ячейках, это не запрещается. Также не запрещается решать теоретические задачи на бумаге, оформлять их аккуратно и разборчиво, и присылать pdf со сканами, вместо того, чтобы набирать в ЛАТЭХ.

## Контрольные вопросы

Ниже приводится список вопросов, с ответами на которые может быть полезно разобраться для комфортного выполнения задания.

### Основные понятия

1. Что такое задачи классификации, кластеризации и регрессии? Какие из них относятся к supervised learning, а какие - к unsupervised?
2. Что такое переобучение и недообучение? Как их можно детектировать?
3. Что такое обучающая и тестовая выборки, кросс-валидация? Как устроена k-fold cross validation?

### Простые методы

1. Как работает kNN в задаче классификации?
2. Как работает kNN с весами объектов в задаче классификации и в задаче регрессии?
3. Как работает наивный байесовский классификатор, в чем заключается его «наивность»?

4. Как приближается исходная зависимость  $y$  от  $x$  в линейной регрессии и как настраиваются веса в ней?

### Python, numpy, scipy, matplotlib

1. Типы данных list, tuple, dict, set, str, unicode, hashable и unhashable типы. Управляющие конструкции в python (циклы, условные операторы), объявление функций. Map и reduce, list comprehensions, генераторы, лямбда-выражения. Чтение и запись в файл в Python.
2. Зачем нужны numpy и scipy? Какой тип данных в numpy используется для работы с многомерными массивами? Отличия в индексации двумерного ndarray и списка списков.
3. Как в scipy решить численно несложную оптимизационную задачу? Какие методы оптимизации в нем представлены?
4. Как по списку значений  $x$  и списку значений  $y$  в этих точках построить график  $y(x)$  в matplotlib?

### Метрики качества в задачах классификации и регрессии

1. Как вычисляются и в каких задачах (классификации/регрессии) применяются метрики: accuracy, precision, recall, F1-measure, ROC-AUC, log loss, MSE, MAE, RMSE?
2. Решается задача бинарной классификации (с двумя классами — 0 и 1), в которой примеры из класса 0 составляют 95% выборки. Какие метрики из перечисленных в предыдущем вопросе предпочтительней использовать?
3. К оценке какой величины для распределения  $y$  при условии  $x$  приводят MSE и MAE?
4. Можно ли при таргетах из множества  $Y = \{0; 1\}$  использовать для оценки  $P(y = 1|x)$  не log loss, а MSE?

### Scikit-learn и pandas

1. Как в sklearn обучить модель на обучающей выборке и получить прогнозы на тестовой?
2. Какие есть средства для измерения качества модели в sklearn? Как посчитать качество в k-fold cross validation?
3. Какие метрики можно использовать в cross\_val\_score из sklearn?
4. Как считать выборку из csv в pandas DataFrame? А как записать DataFrame в файл? Как указывать при чтении/записи кодировку, используемые разделители, наличие/отсутствие заголовков у колонок?
5. Как по списку значений  $x$  и списку значений  $y$  в этих точках построить график  $y(x)$  в matplotlib?

## 1 Метод k ближайших соседей

*10% баллов за задание, оценочное время выполнения: 20 минут*

Сгенерируйте обучающую выборку из описанных двумя признаками объектов нескольких классов и визуализируйте разделяющие поверхности, получаемые при решении задачи классификации методом k ближайших соседей для разных k. Попробуйте подобрать оптимальное значение количества соседей k с помощью 5-fold cross-validation, построив график зависимости accuracy в кросс-валидации от k.

## 2 Наивный байесовский классификатор

20% баллов за задание, оценочное время выполнения: 40 минут

Загрузите датасеты `digits` и `breast_cancer` из `sklearn.datasets`. Выведите несколько строчек из обучающих выборок и посмотрите на признаки. С помощью `sklearn.model_selection.cross_val_score` с настройками по умолчанию и вызова метода `mean()` у возвращаемого этой функцией `numpy.ndarray`, сравните качество работы наивных байесовских классификаторов на этих двух датасетах. Для сравнения предлагается использовать `BernoulliNB`, `MultinomialNB` и `GaussianNB`. Насколько полученные результаты согласуются с вашими ожиданиями?

Два датасета, конечно, еще не повод делать далеко идущие выводы, но при желании вы можете продолжить исследование на других выборках (например, из UCI репозитория).

Ответьте (прямо в `ipynb` блокноте с вашими экспериментами) на вопросы:

1. Каким получилось максимальное качество классификации на датасете `breast_cancer`?
2. Каким получилось максимальное качество классификации на датасете `digits`?
3. Какие утверждения из приведенных ниже верны?
  - (a) На вещественных признаках лучше всего сработал наивный байесовский классификатор с распределением Бернулли
  - (b) На вещественных признаках лучше всего сработал наивный байесовский классификатор с мультиномиальным распределением
  - (c) Мультиномиальное распределение лучше показало себя на выборке с целыми неотрицательными значениями признаков
  - (d) На вещественных признаках лучше всего сработало нормальное распределение

## 3 Метрики в задаче регрессии

40% баллов за задание, оценочное время выполнения: 120 минут

Сгенерируйте датасет из 500 точек на плоскости, для которых  $y = 0.5x + 1 + \epsilon$ , где  $\epsilon$  распределено нормально с матожиданием 0 и дисперсией 0.2.

1. Визуализируйте выборку.
2. Восстановите по выборке зависимость  $y(x)$ , считая, что зависимость имеет вид  $y = kx + b$ , и минимизируя MSE на обучающей выборке, воспользовавшись `scipy.optimize.minimize`. Визуализируйте восстановленную прямую.
3. Добавьте теперь в выборку 75 точек, для которых  $y = -1 + \epsilon$ , а  $x$  принимает различные значения из того же диапазона, что и у уже имевшихся точек в обучающей выборке. По новой расширенной выборке снова попробуйте восстановить зависимость  $y(x) = kx + b$  двумя способами: минимизируя MSE и минимизируя MAE. Визуализируйте полученные прямые.
4. На основе полученных графиков сделайте вывод об устойчивости моделей, оптимизирующих MSE и MAE к выбросам.

## 4 Теоретические задачи

30% баллов за задание

### 4.1 Наивный байес и центроидный классификатор

Покажите, что если в наивном байесовском классификаторе классы имеют одинаковые априорные вероятности, а плотность распределения признаков в каждом классе имеет вид  $P(x^{(k)}|y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x^{(k)} - \mu_y)^2}{2\sigma^2}}$ ,  $x^{(k)}, k = 1, \dots, n$  — признаки объекта  $x$  классификация сводится к отнесению объекта  $x$  к классу  $y$ , центр которого  $\mu_y$  ближе всего к  $x$ .

## 4.2 ROC-AUC случайных ответов

Покажите, что «треугольный ROC-AUC» (см. лекцию 2) в случае, когда классификатор дает случайные ответы —  $a(x) = 1$  с вероятностью  $p$  и  $a(x) = 0$  с вероятностью  $1 - p$ , будет в среднем равен 0.5, независимо от  $p$  и доли класса 1 в обучающей выборке.

## 4.3 Ошибка 1NN и оптимального байесовского классификатора

Утверждается, что метод одного ближайшего соседа асимптотически (при условии, что максимальное по всем точкам выборки расстояние до ближайшего соседа стремится к нулю) имеет матожидание ошибки не более чем вдвое больше по сравнению с оптимальным байесовским классификатором (который это матожидание минимизирует).

Покажите это, рассмотрев задачу бинарной классификации. Достаточно рассмотреть вероятность ошибки на фиксированном объекте  $x$ , т.к. матожидание ошибок на выборке размера  $V$  будет просто произведением  $V$  на эту вероятность. Байесовский классификатор ошибается на объекте  $x$  с вероятностью:

$$E_B = \min\{P(1|x), P(0|x)\}$$

Условные вероятности будем считать непрерывными функциями от  $x \in R^m$ , чтобы иметь возможность делать предельные переходы. Метод ближайшего соседа ошибается с вероятностью:

$$E_N = P(y \neq y_n)$$

Здесь  $y$  - настоящий класс  $x$ , а  $y_n$  - класс ближайшего соседа  $x_n$  к объекту  $x$  в предположении, что в обучающей выборке  $n$  объектов, равномерно заполняющих пространство.

Докажите исходное утверждение, выписав выражение для  $E_N$  (принадлежность к классам 0 и 1 для объектов  $x$  и  $x_n$  считать независимыми событиями) и осуществив предельный переход по  $n$ .