ТЕХНОСФЕРА

Лекция 3
Методы снижения размерности
Отбор признаков

Владимир Гулин

https://goo.gl/p8BjOD

20 февраля 2016 г.

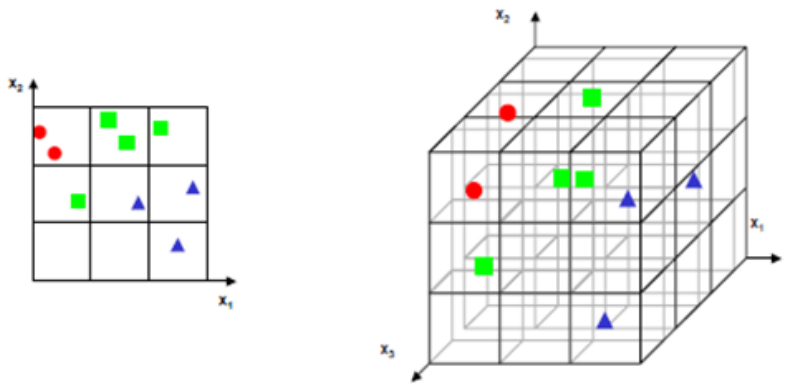
План лекции

Мотивация

- Визуализация
- Скорость обучения
- Качество обучения
- Экономия при эксплуатации
- Понимание данных и гибкость построения новых моделей

Проклятие размерности (curse of dimensionality)

- Сложность вычислений возрастает экспоненциально
- Требуется хранить огромное количество данных
- Большое число признаков являются шумными
- В линейных классификаторах увеличение числа признаков приводит к мультиколлинеарности и переобучению.
- Для метрических классификаторов (в пространсвах с l_p нормой) согласно закону больших чисел расстояния становятся неинформативны.

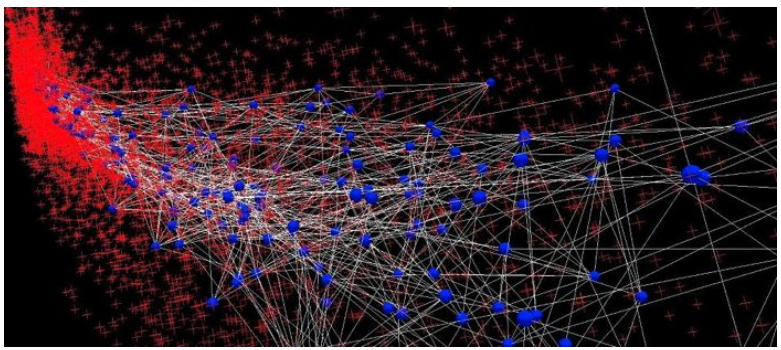


Подходы к снижению размерности

Feature Extraction

Data space $\rightarrow$ Feature space

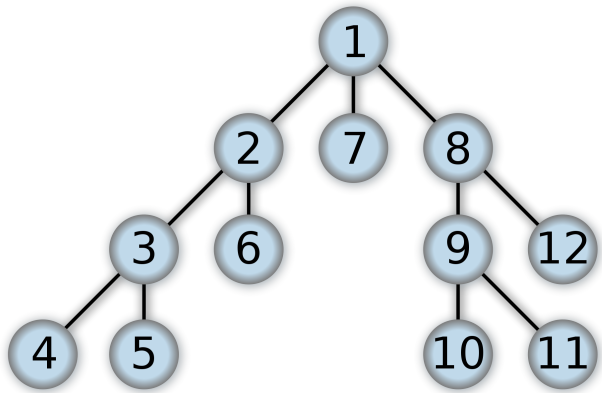
Пространство данных может быть представлено сокращенным количеством “эффективных” признаков



Feature Selection

Data space $\rightarrow$ Data subspace

Отбирается некоторое подмножество наиболее “полезных” признаков



Задача отбора признаков

Feature Selection

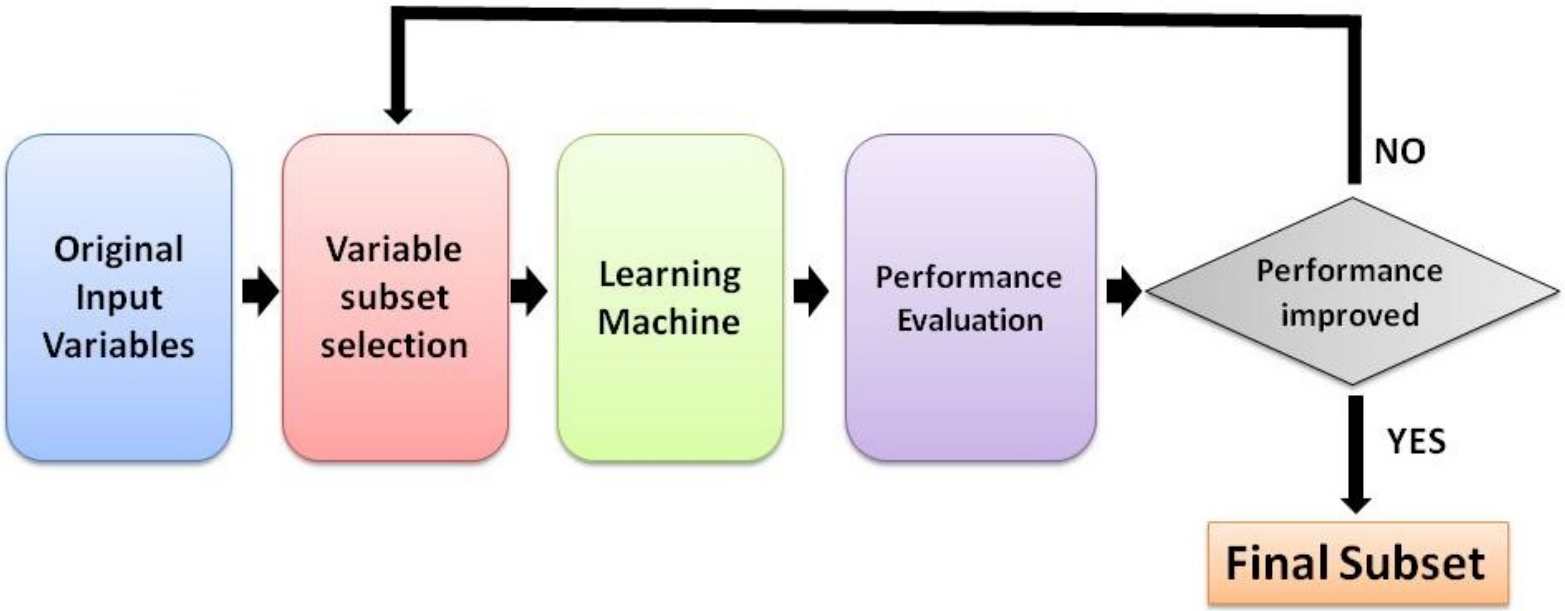
Дано. N обучающих D-мерных объектов $\mathbf{x}_i \in \mathcal{X}$, образующих тренировочный набор данных (training data set) $\mathbf{X}$, а также каждому $\mathbf{x}_i$ соответсвует метка $y_i \in \mathcal{R}$.

Найти. Найти подмножество признаков F исходного признакового пространства $\mathcal{F} = \{f_1, f_2, \ldots, f_D\}$, содержащее наиболее "информативные" признаки.

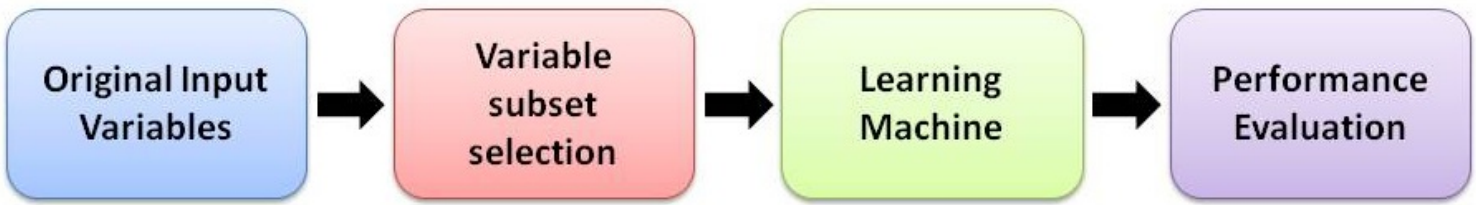
Классификация методов отбора признаков:

- Wrapper methods
- Filter methods
- Embedded methods

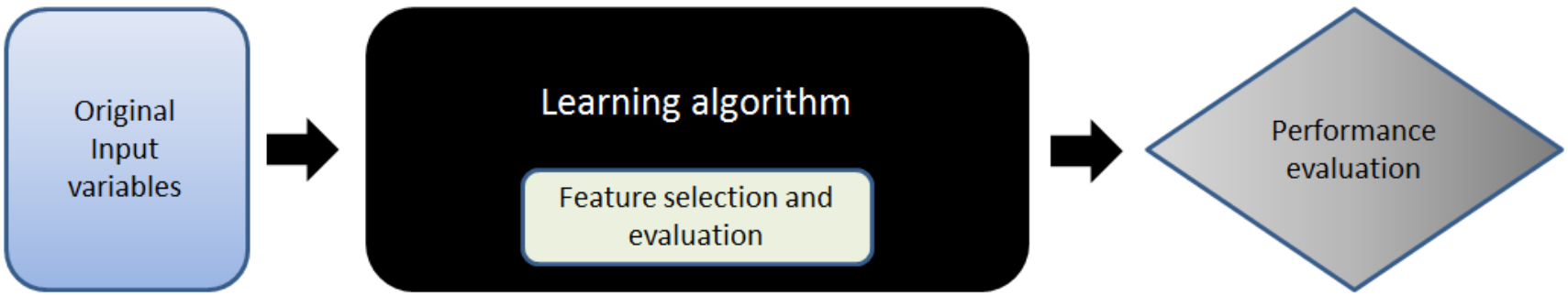
Wrapper methods



Filter methods



Embedded methods



Выбор модели обучения

Дано:

X - пространство объектов; Y - множество ответов;
$X^N = (x_i, y_i)_{i=1}^N$ - обучающая выборка, $y_i = f(x_i)$;
$a_k = \{a : X \to Y\}$ - типы моделей, $k \in K$;
$\mu_k : (X \times C)^N \to a_k$ - методы обучения, $k \in K$.

Найти:

Метод обучения μ с лучшей обобщающей способностью

Частные случаи:

- Выбор метода обучения μ - оптимизация параметров модели a
- Model selection a
- Feature selection:
  $F = \{f_{j_1}, \ldots, f_{j_d}\}$ такое, что метод обучения μ использует только признаки из подмножества F.

Как оценить качество

$L(a, x)$ - функция потерь (loss function) алгоритма a на примере x;

$J(a, X^N) = \frac{1}{N} \sum_{i=1}^{N} L(a, x_i)$ - функционал эмпирического риска для a на выборке X^N;

Внутренний критерий:

Оцениваем качество на обучающей выборке

$$J_\mu(X^N) = J(\mu(X^N), X^N).$$

Однако получаем смещенную оценку

Внешний критерий:

Оцениваем качество на данных, не участвующих в процессе обучения, например, на проверочной выборке X^Q

$$J_\mu(X^N, X^Q) = J(\mu(X^N), X^Q).$$

Оценка зависит от разбиения $X^Z = X^N \sqcup X^Q$

Кросс-валидация

Усреднение оценок внешних критериев по заданному T - множеству разбиений $X^Z = X_t^N \sqcup X_t^Q,\ t = 1, \ldots, T$

$$CV(\mu, X^Z) = \frac{1}{|T|} \sum_{t=1}^{T} J_\mu(X_t^N, X_t^Q)$$

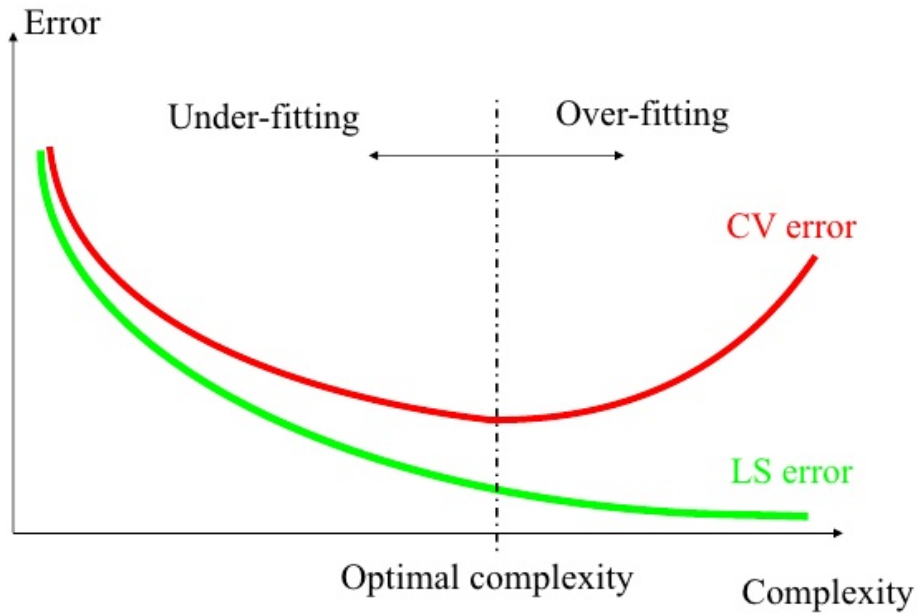
Варианты кросс-валидации

1. Случайное множество разбиений
2. Полная кросс-валидация. Используем все возможные подмножества размера Q из Z. Сколько их?
3. Скользящий контроль (Leave one out). $Z = 1$
4. Проверка по блокам
5. Многократная проверка по блокам

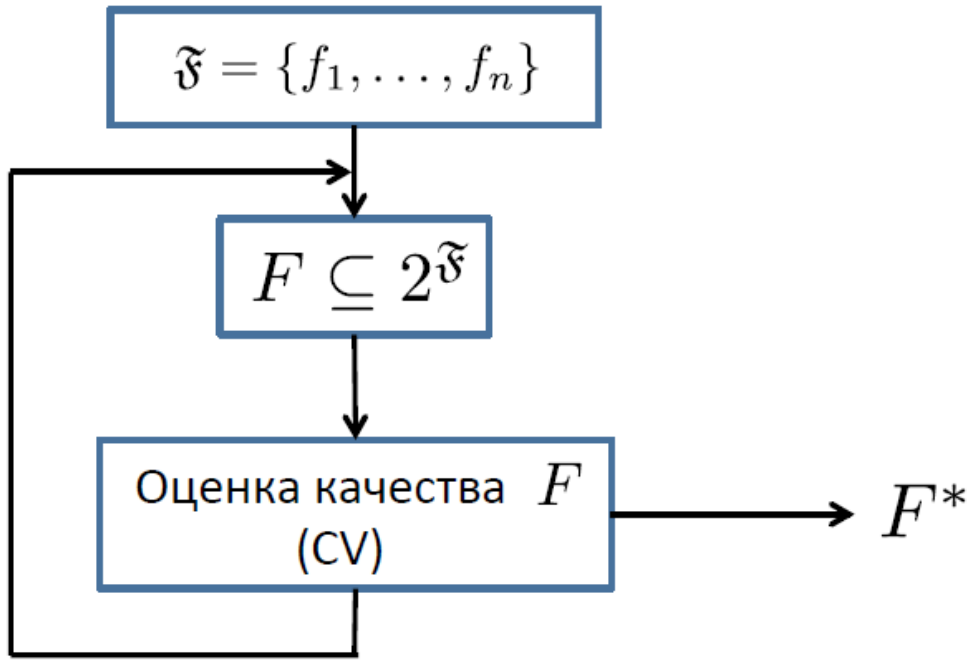
Задача отбора признаков по внешнему критерию

$F = \{f_{j_1}, \ldots, f_{j_d}\}$ - множество признаков такое, что метод обучения μ использует только признаки из подмножества F.
$J(F) = J(\mu_F, X^N)$ - выбранный внешний критерий

$$J(F) \to min$$



Отбор признаков “в лоб”



- Экспертный подход
- Full Search (NP hard)
- Жадные алгоритмы (Forward selection, Backward elimination, Bidirectional elimination etc.)

Жадные алгоритмы отбора признаков

Forward selection

```
1  function forwardselection(F, J, n):
2      # F - original feature set
3      # J - external criterion
4      # n - parameter
5      initialize F_0 = {} # empty set
6      initialize Q = J(F_0) # compute score
7      for j in 1..D:
8          fbest = find_best_feature(J, F_j-1, F)
9          F_j = add_new_feature(F_j-1, fbest) # add feature
10         if J(F_j) < Q:
11             jbest = j
12             Q = J(F_j) # save best
13         if j - jbest >= n:
14            return F_jbest
```

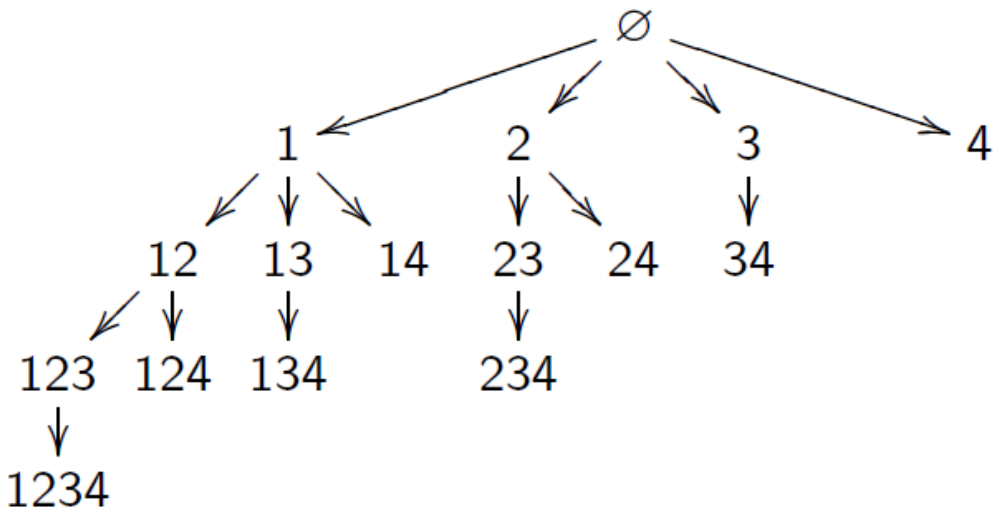
Backward elimination

- Все аналогично. Только ислючаем

Жадные алгоритмы отбора признаков

DFS. Основные идеи:

- Избегаем повторов при переборе
- Если подмножество признаков бесперспективно, то не будем пытаться его дальше наращивать.



Оценка бесперспективности:

$$\exists j: \quad J(F) \geq \eta J(F_j^*), \quad |F| \geq j + n$$

Жадные алгоритмы отбора признаков

Итоги

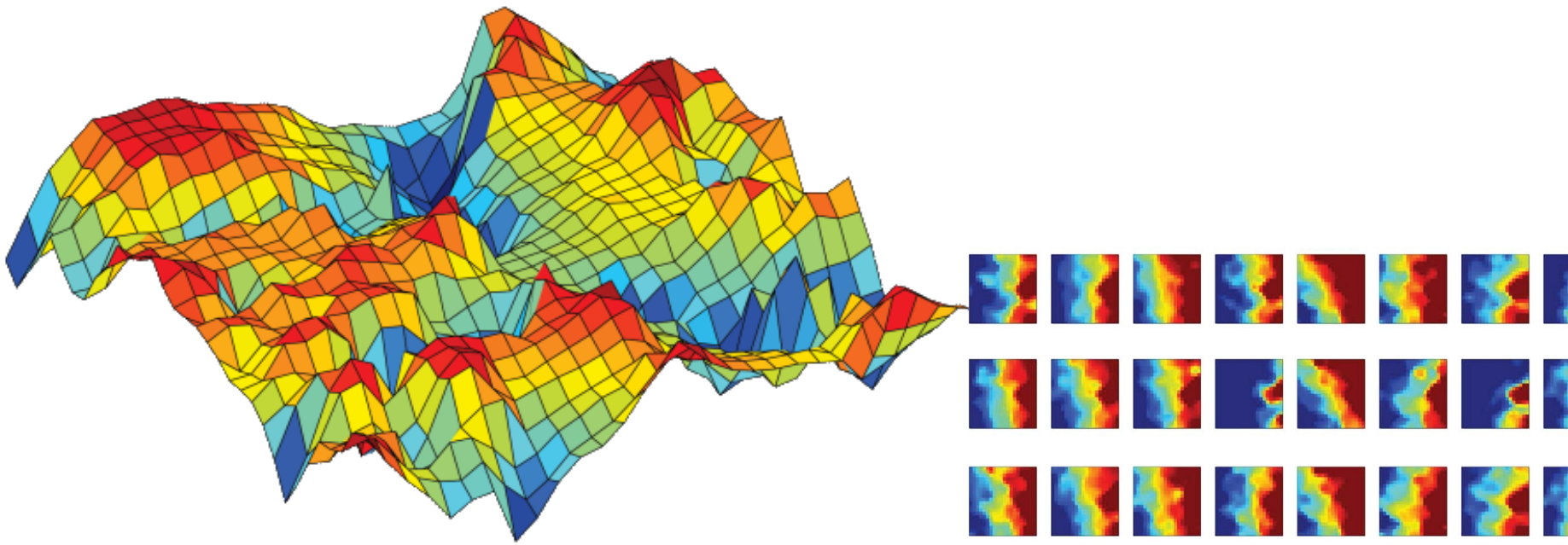
- Не все признаки “полезны”
- Отбор признаков проводится по внешним критериям (CV)
- Для сокращения перебора хороши любые эвристики
- Предполагаем, что перебор по подмножествам устойчив
- НАДО ПЕРЕОБУЧАТЬ АЛГОРИТМ

Теоретико-информационные методы. Интуиция

А можно ли что-то сказать о фиче не применяя алгоритм обучения?

- Не берем фичи с малой дисперсией
- Не берем фичи “не похожие” на ответ
- Не берем фичи похожие, на те, которые уже взяли

Теоретико-информационные методы. Интуиция



“Нерелевантность” признака

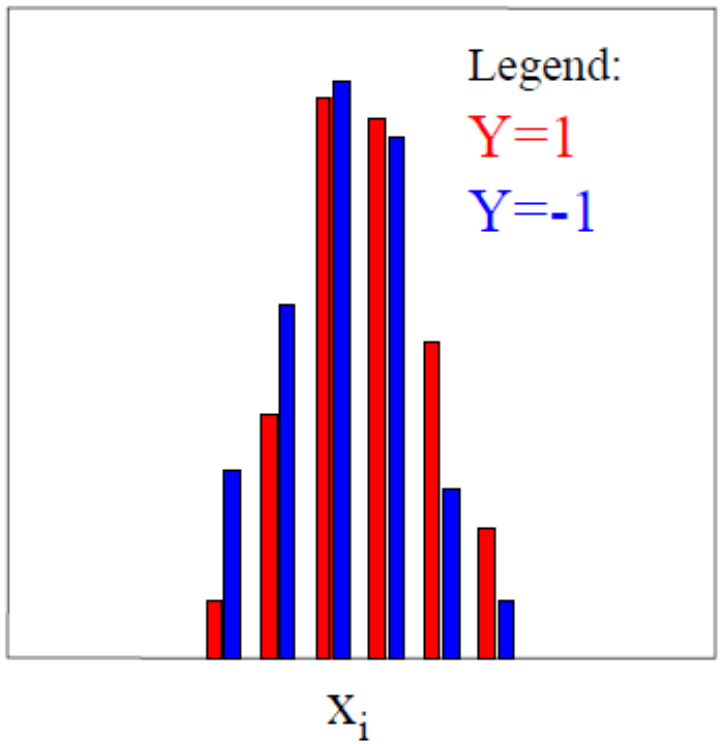
Рассмотрим задачу бинарной классификации

$X \in \mathcal{R}^D, \quad Y \in \{-1, +1\}$

$$P(X_i, Y) = P(X_i)\, P(Y)$$

$$P(X_i | Y) = P(X_i)$$

$$P(X_i | Y=1) = P(X_i | Y=-1)$$



“Нерелевантность” признака

Рассмотрим задачу многоклассовой классификации
$X \in \mathcal{R}^D, \quad Y = \{y_1, y_2, \dots y_k\}$

$$\chi^2 = \sum_{ij} \frac{(M_{ij} - m_{ij})^2}{m_{ij}}, \quad m_{ij} = \frac{1}{N} M_{i\cdot} M_{\cdot j}$$

где $M_{ij}-$ количество примеров в датасете с $f = f_i, y = y_j$.

- Какое значение соответсвует релевантности признака?

“Нерелевантность” признака

Рассмотрим задачу бинарной классификации
$X \in \mathcal{R}^D, \quad Y \in \{-1, +1\}$

Отношение “сигнал-шум”

$$\mu(X, Y) = \frac{\mu(y_+) - \mu(y_-)}{\sigma(y_+) + \sigma(y_-)}$$

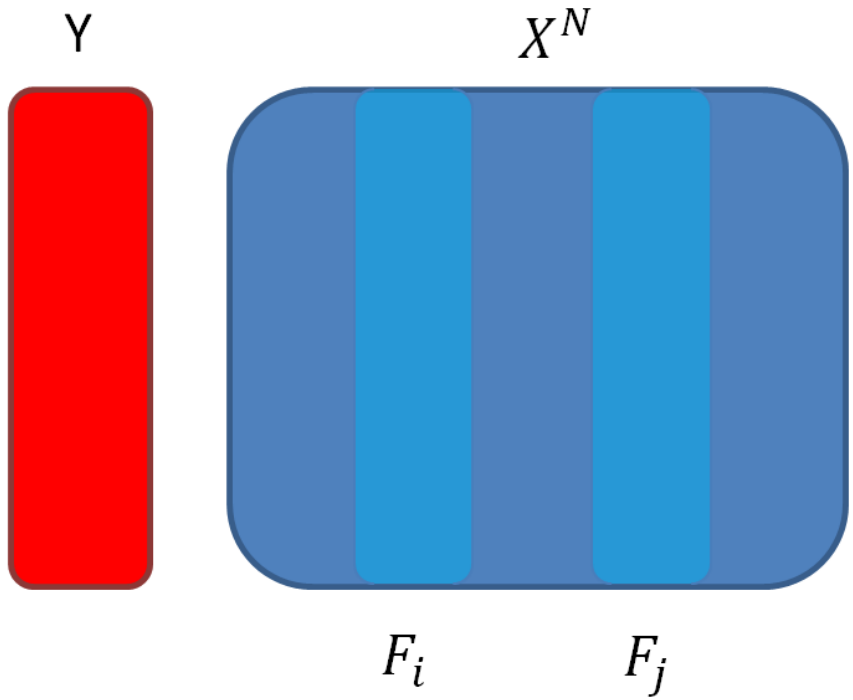
$$\mu(y_+) = \frac{1}{N_+} X(: |y_+), \quad \mu(y_-) = \frac{1}{N_-} X(: |y_-)$$

$$\sigma(y_+) = \sqrt{\frac{1}{N_+ - 1} \sum_{j \in N_+} (X(j|y_+) - \mu(y_+))^2},$$

$$\sigma(y_-) = \sqrt{\frac{1}{N_- - 1} \sum_{j \in N_-} (X(j|y_-) - \mu(y_-))^2}$$

Методы основанные на корреляции/взаимной информации

Интуиция



- Хотим найти признаки “похожие” на ответы
- Хотим, чтобы признаки сами между собой были “не похожи”

Методы основанные на корреляции/взаимной информации

Коэффициент корреляции

$$r(X,Y) = \frac{\sum_x (x-\bar{x}) \sum_y (y-\bar{y})}{\sqrt{\sum_x (x-\bar{x})^2} \sqrt{\sum_y (y-\bar{y})^2}}$$

- Correlation feature selection (cfs)

Взаимная информация

$$I(X,Y) = \sum_x \sum_y p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) = D_{KL}(p(X,Y)||p(X)p(Y))$$

- Minimum Redundancy maximum Relevance (mRMR)

CFS

$$r_{yf} = \frac{\sum_y (y - \bar{y}) \sum_f (f - \bar{f})}{\sqrt{\sum_y (y - \bar{y})^2} \sqrt{\sum_f (f - \bar{f})^2}}$$

$$r_{f_i f_j} = \frac{\sum_{f_i} (f_i - \bar{f}_i) \sum_{f_j} (f_j - \bar{f}_j)}{\sqrt{\sum_{f_i} (f_i - \bar{f}_i)^2} \sqrt{\sum_{f_j} (f_j - \bar{f}_j)^2}}$$

Критерий *CFS*:

$$CFS = \max_{F_d} \left[\frac{r_{yf_1} + r_{yf_2} + \ldots + r_{yf_d}}{\sqrt{d + 2(r_{f_1 f_2} + r_{f_1 f_3} + \ldots r_{f_{d-1} f_d})}} \right]$$

Вопрос:

- Оцените сложность алгоритма CFS

mRMR

Идея

- Будем отбирать признаки, которые имеют наибольшую взаимную информацию с ответами
- Будем штрафовать признаки за избыточность, в контексте уже отобранных фичей

$$Relevance(F, y) = \frac{1}{|F|} \sum_{f_i \in F} I(f_i, y)$$

$$Redundancy(F) = \frac{1}{|F|^2} \sum_{f_i, f_j \in F} I(f_i, f_j)$$

Тогда критерий mRMR имеет вид:

$$mRMR = \max_F (Relevance(F, y) - Redundancy(F))$$

Embedded methods

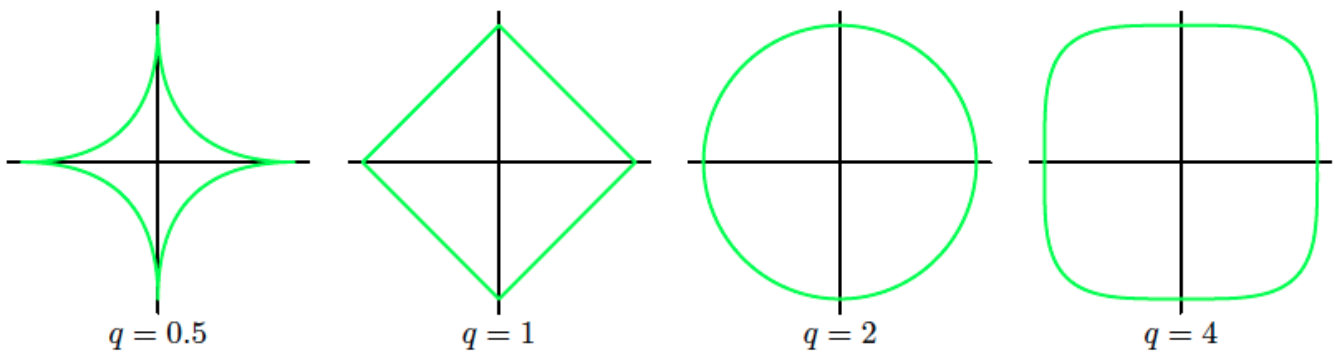
Linear Regression with Regularization

$a(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ - линейная модель

$$J(a, X^N) = \sum_{i=1}^{N}(a(x_i) - c_i)^2$$

Регуляризация

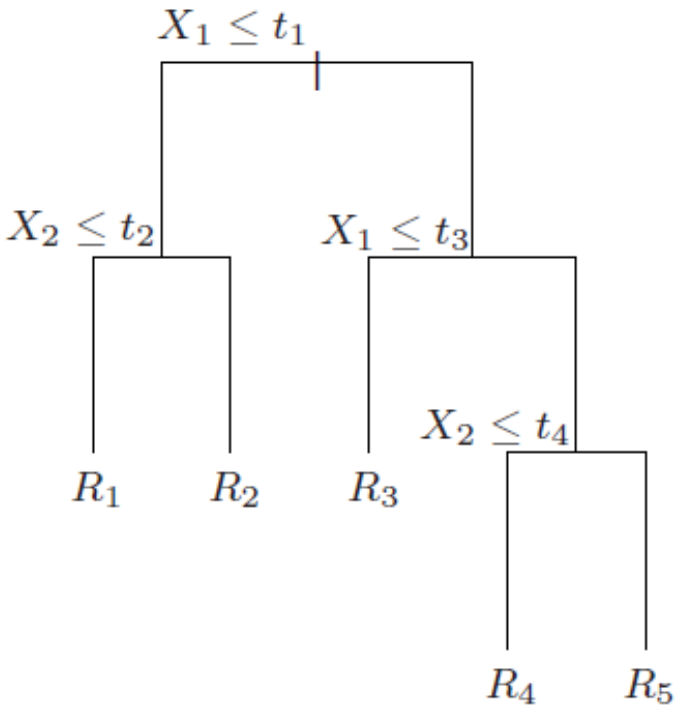
$$J(a, X^N) = \sum_{i=1}^{N}(a(x_i) - c_i)^2 + \lambda\|\mathbf{w}\|_q$$

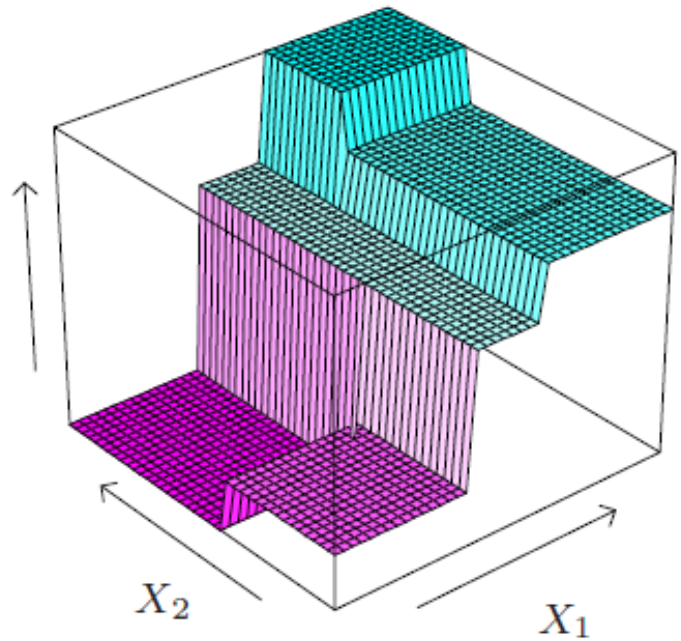


Embedded methods

Decision Trees with pruning

Деревья решений обладают бесконечной VC размерностью

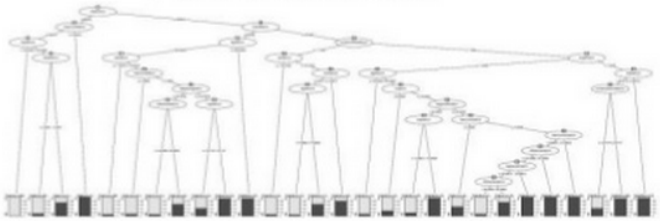


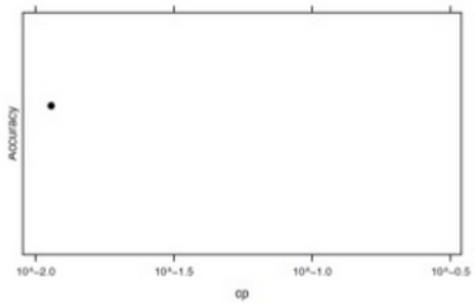


Embedded methods

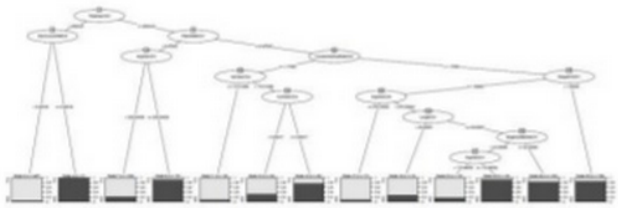
Decision Trees with pruning

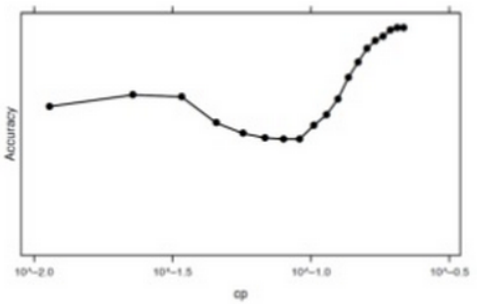
Full Decision Tree



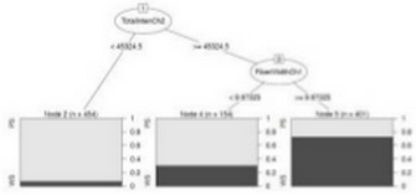


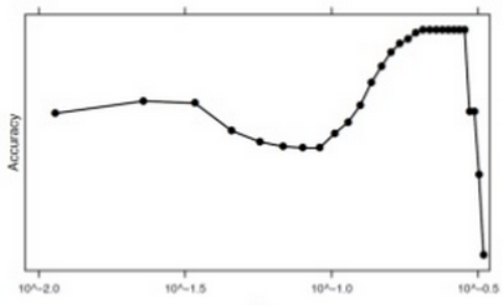
Start Pruning





Too Much!!!



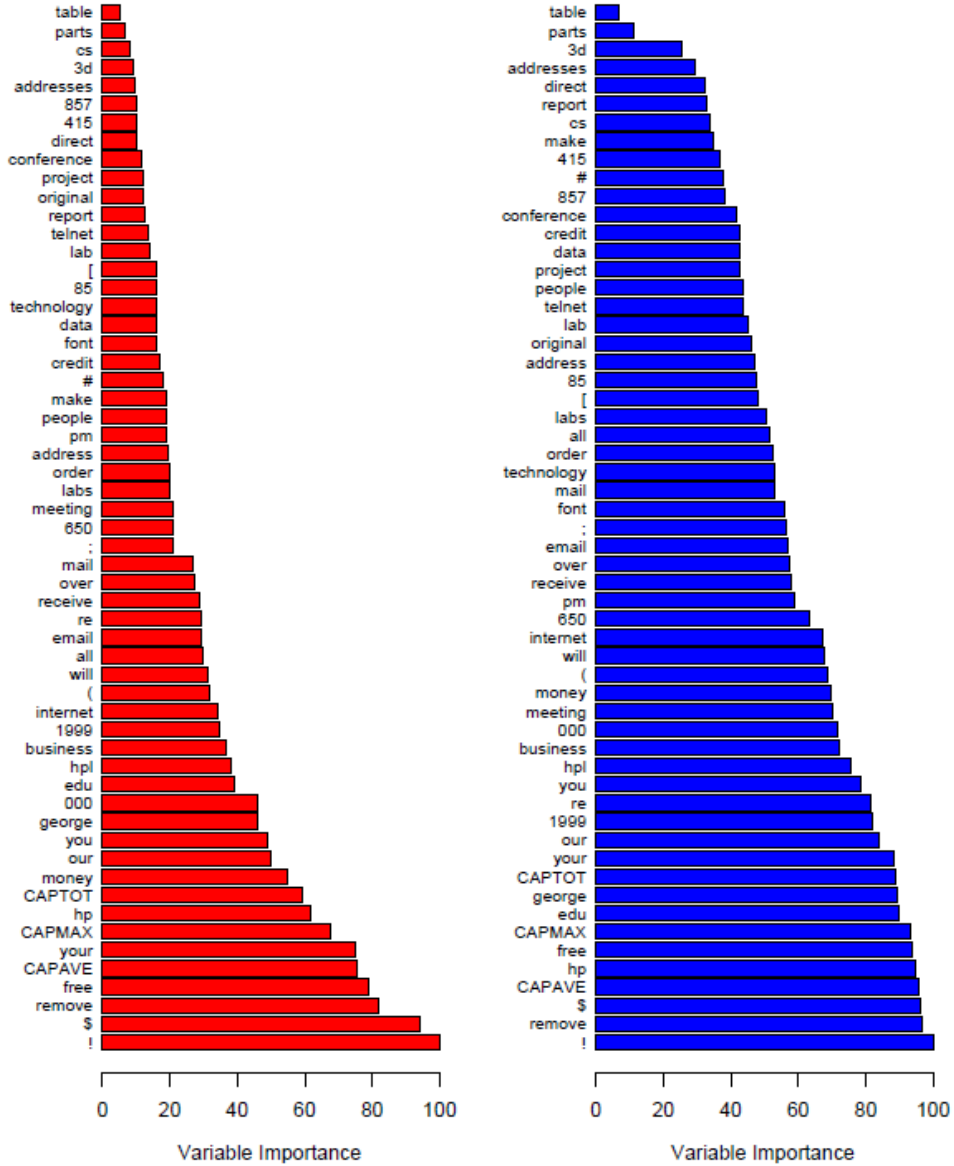


Embedded methods

Ранжирование признаков

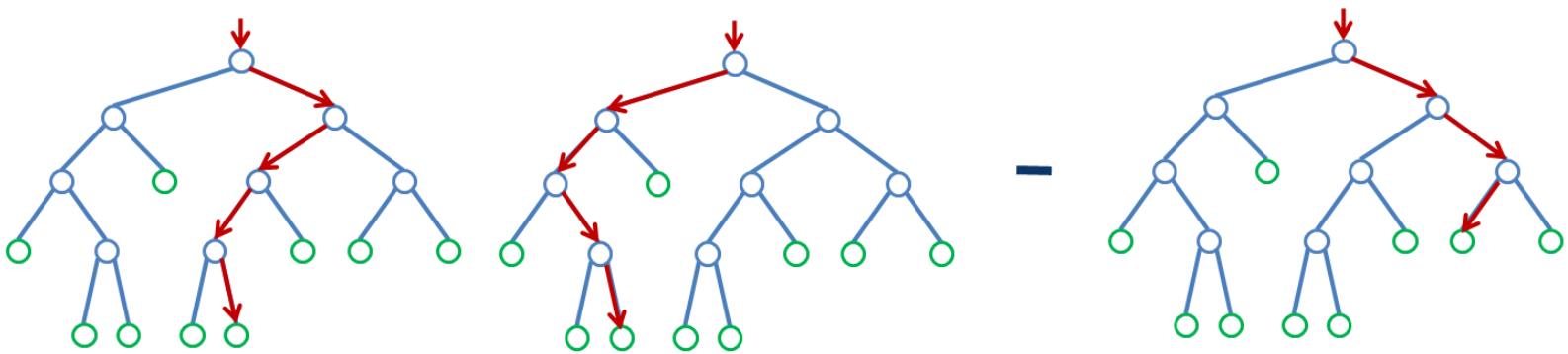
- Как понять какие признаки более важны?
- В линейных моделях?
- В нелинейных моделях (деревьях)?
- В ансамблях?

Feature impotance for decision tree



Feature impotance for ensembles

- Просуммируем по всем базовым моделям
- Как быть с нелинейными композициями (stacking)?



Wrapper methods

Выводы

- ✔ Непосредственно проводит процедуру выбора модели. Что зачастую позволяет найти “почти” оптимальное подмножество
- ✖ Требует построения алгоритма для каждого подмножества признаков. ОООЧЕНЬ дорого с вычислительной точки зрения.

Filter methods

Выводы

- ✔ Легко масштабируются для датасетов высокой размерности
- ✔ Дешевы с вычислительной точки зрения
- ✔ Можно применить метод только один раз и затем использовать для большого числа алгоритмов машинного обучения
- ✖ Не “ловит” feature interaction

Embedded methods

Выводы

- ✔ Менее вычислительно затратны, чем wrapper методы
- ✖ Результат сильно зависит от используемого алгоритма обучения
- ✖ Результат сильно зависит от датасета

Отбор признаков в реальной жизни. Оценка качества признаков

Как понять, что новая фича реально работает и приносит профит?

Отбор признаков в реальной жизни. Оценка качества признаков

Как понять, что новая фича реально работает и приносит профит?

- Надо проверить статистическую гипотезу о том, что с этой фичой качество выше, чем без нее
- На практике это выраждается в многократную кроссвалидацию по блокам со сравнением scora

Домашнее задание №2

Требуется: Реализовать embedded метод отбора признаков для своего варианта из дз 1. Провести сравнительный анализ своего метода с каждым из подходов, описанных на лекции (т.е. нужно минимум реализовать один wrapper и один filter метод). Построить графики.

Критерии анализа

1. Быстродействие
2. Качество полученных моделей, в зависимости от количества признаков
3. ...

Вопросы