Занятие 7. Нелинейные методы. Отбор признаков

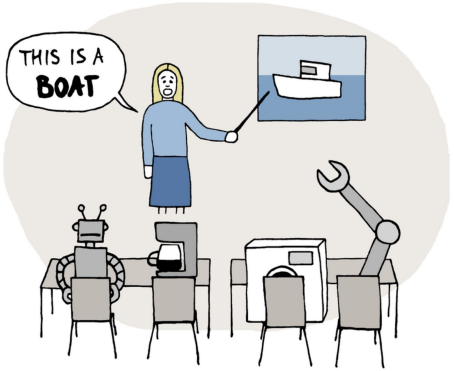
Колмагоров Евгений
ml.hse.dpo@yandex.ru

25 ноября 2024

План лекции

1. Наивный байесовский классификатор
2. Метод ближайшего соседа
3. Отбор признаков

Наивный байесовский классификатор



Теорема Байеса

Прежде чем разбирать алгоритм машинного обучения, вспомним одну из центральных теорем теории вероятности – теорему Байеса.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Томас Байес

Необходимые термины

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

В данной формуле:

- P(A) – априорная вероятность некоторого события, до наблюдения события B
- P(A|B) – апостериорная вероятность события A после наблюдения события B
- P(B|A) – вероятность наступления события B при наступлении события A
- P(B) – полная вероятность события B

Связь с машинным обучением

Может возникнуть вполне логичный вопрос, а какая тут связь теоремы байеса с машинным обучением?

Теорема байеса носит универсальный характер, и за под событиями A и B могут иметься в виду объекты произвольной природы. Так в случае задачи классификации за “событием” A находится метка класса Y некоторого объекта, а за “событием” B – признаки X этого объекта.

$$P(Y|X) = \frac{P(X|Y) \cdot P(Y)}{P(X)}$$

Более подробный вариант

$$P(Y|X) = \frac{P(X|Y)\cdot P(Y)}{P(X)}$$

Распишем заданную формулу более подробно в понятиях меток класса $Y = \{y_1, y_2, \ldots, y_K\}$ и признаков объекта $x = (x_1, x_2, \ldots, x_d)$:

$$P(y_j|x_1, x_2, \ldots, x_d) = \frac{P(x_1,x_2,\ldots,x_d|y_j)\cdot P(y_j)}{P(x_1,x_2,..,x_d)}$$

Таким образом по данной формуле можно для каждого класса находить вероятность принадлежности некоторого объекта к заданному классу.

Решающее правило

В качестве ответа будем выдавать класс с наибольшей вероятностью:

$$y = argmax_{i=1,\ldots,K} \frac{P(x_1, x_2, \ldots, x_d | y_i) \cdot P(y_i)}{P(x_1, x_2, .., x_d)}$$

Избавление от знаменателя

Так как в формуле предсказания нас интересует не точное значение вероятности, а то какой класс имеет наибольшее значение вероятности, то знаменатель можно не вычислять:

$$P(y_j|x_1, x_2, \ldots, x_d) = argmax_{i=1,\ldots,K} \frac{P(x_1,x_2,\ldots,x_d|y_i)\cdot P(y_i)}{P(x_1,x_2,..,x_d)} =$$

$$= argmax_{i=1,\ldots,K} P(x_1, x_2, \ldots, x_d|y_i) \cdot P(y_i)$$

Всё готово, но есть одно но…..

Оценки вероятностей P будем проводить на основе обучающего множества X. Но есть проблема, что оценить вероятность для заданного класса $P(x_1, x_2, \ldots, x_d | y_j)$ не всегда возможно, так как может не найтись объекта в обучении с заданными значениями признаков $x_1, x_2, \ldots, x_d$

$$P(x_1, x_2, \ldots, x_d | y_j) - ?$$

Гипотеза “наивности”

Будем считать, что каждый из факторов x_1, …, x_d не зависит друг от друга, тогда по формуле независимых событий:

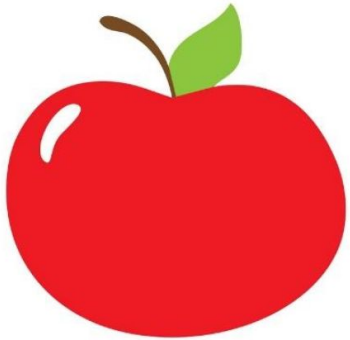
$$P(x_1, x_2, \ldots, x_d|y_j) = P(x_1|y_j) \cdot P(x_2|y_j) \cdot \ldots \cdot P(x_d|y_j)$$

Допущение о том, что каждый из признаков независим, является достаточно сильным упрощением. Например, в задаче предсказания заболеваемости признаки рост и вес, имеют достаточно сильную связь друг с другом.

Модельный пример

Будем считать фрукт яблоком, если он:

- Круглый
- Красный
- Его диаметр составляет порядка 8 см



Тогда вероятность считать некоторый фрукт яблоком:

$$P(y = \text{яблоко} | X = (\text{круглый}, \text{красный}, \text{диаметр})) \propto$$

$$[P(\text{красный}|\text{яблоко}) \cdot P(\text{круглый}|\text{яблоко}) \cdot P(\text{диаметр}|\text{яблоко})] \cdot P(\text{яблоко})$$

Вопрос: Какие могут встретиться проблемы при большом количестве признаков в произведении?

Более сложный пример

Пример: попробуем предсказать состоится ли игра – Play Golf – по данным о погоде:

- Outlook
- Temperature
- Humidity
- Windy

	Outlook	Temperature	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	Normal	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	No
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	Yes
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	High	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Вычисление статистик

Посчитаем $P(x_i \mid Yes)$ и $P(x_i \mid No)$ для каждого признака и априорную вероятность $P(y)$

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
False	6	2	6/9	2/5
True	3	3	3/9	3/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Вычисление статистик

$$P(Yes|Humidity = High, Outlook = Sunny, Wind = False, Temperature = Hot) \propto$$
$$\propto P(High|Yes) \cdot P(Sunny|Yes) \cdot P(False|Yes) \cdot P(Hot|Yes) \cdot P(Yes) =$$
$$= \frac{3}{9} \cdot \frac{2}{9} \cdot \frac{6}{9} \cdot \frac{2}{9} \cdot \frac{9}{14} = \frac{648}{91854} \approx 0.007$$

$$P(No|Humidity = High, Outlook = Sunny, Wind = False, Temperature = Hot) \propto$$
$$\propto P(High|No) \cdot P(Sunny|No) \cdot P(False|No) \cdot P(Hot|No) \cdot P(No) =$$
$$= \frac{4}{5} \cdot \frac{3}{5} \cdot \frac{2}{5} \cdot \frac{2}{5} \cdot \frac{5}{14} = \frac{240}{8750} \approx 0.027$$

Вероятность того, что игра не состоится при заданных погодных условия выше чем она состоится

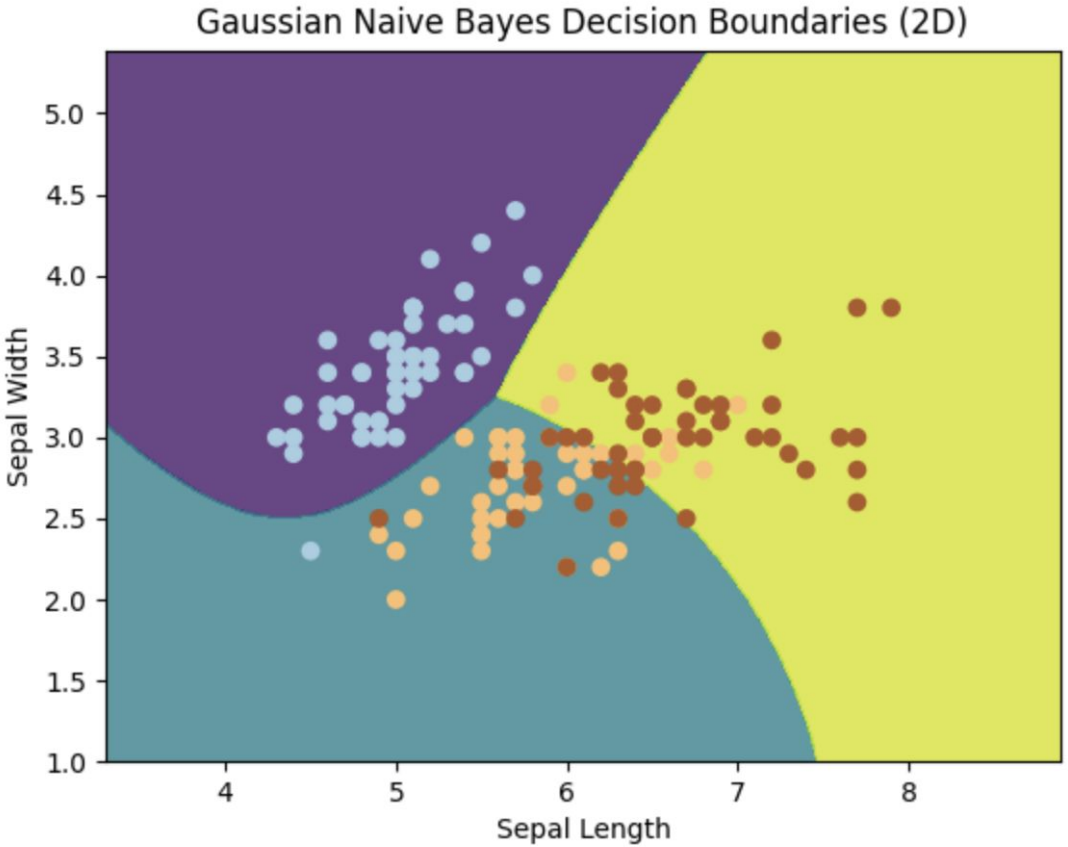
Наивный байес для численных признаков

Численные признаки пытаются приблизить нормальным распределением:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

μ, σ – вычисляются по выборке методом максимального правдоподобия.

Визуализация работы



Зона применимости

Плюсы:

- Простой и быстрый алгоритм классификации
- В случае, если выполняется предположение о независимости признаков, классификатор показывает очень высокое качество работы

Минусы:

- В случае если в тестовых данных присутствует категория, не встречающаяся в обучении, модель присвоить нулевую вероятность

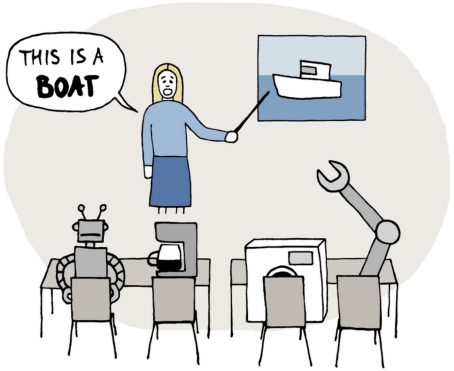
Сглаживание лапласа

Для решения проблемы нулевой вероятности $P(x_i|y_j)$ делают поправку в вычислении истинной вероятности:

$$P(x_i|y_j) = \frac{|\{k: x^k = x_i\}| + \alpha}{|\{k: y^k = y_j\}| + \alpha \cdot K}$$

Где K – количество признаков

Метод ближайшего соседа

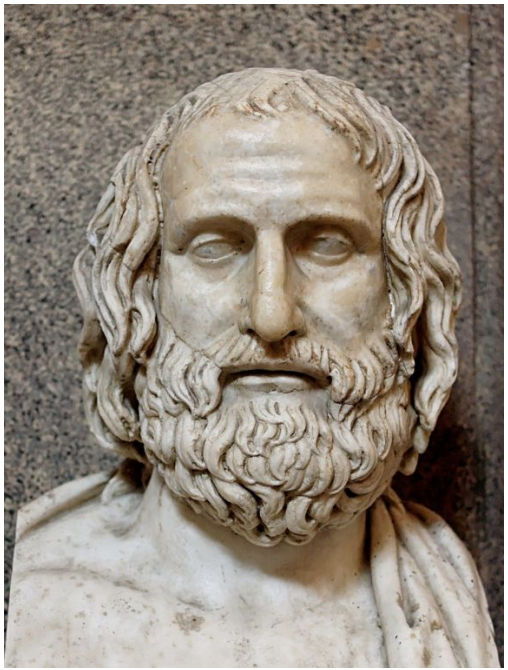


Основополагающий принцип

Принцип, по которому работает алгоритм ближайших соседей, был сформулирован ещё в глубокой древности греческим поэтом Еврипидом.

Данный принцип формулируется так:

"Скажи мне, кто твой друг, и скажу кто ты"

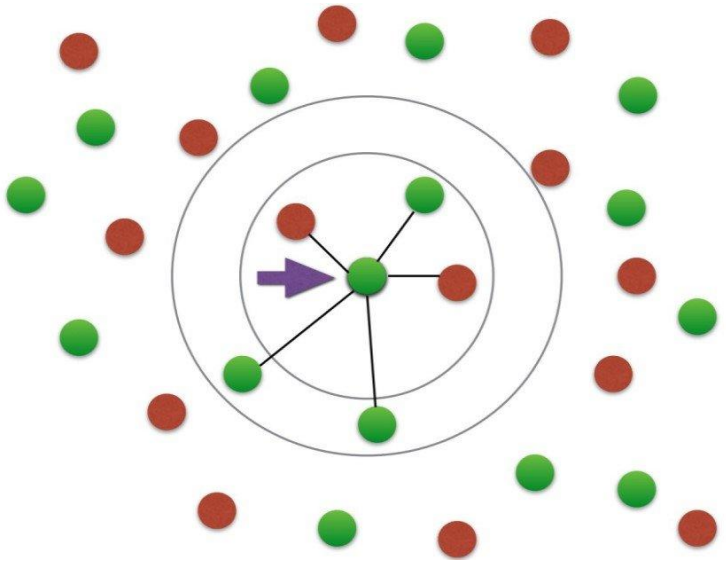


Еврипид

Интерпретация для машинного обучения

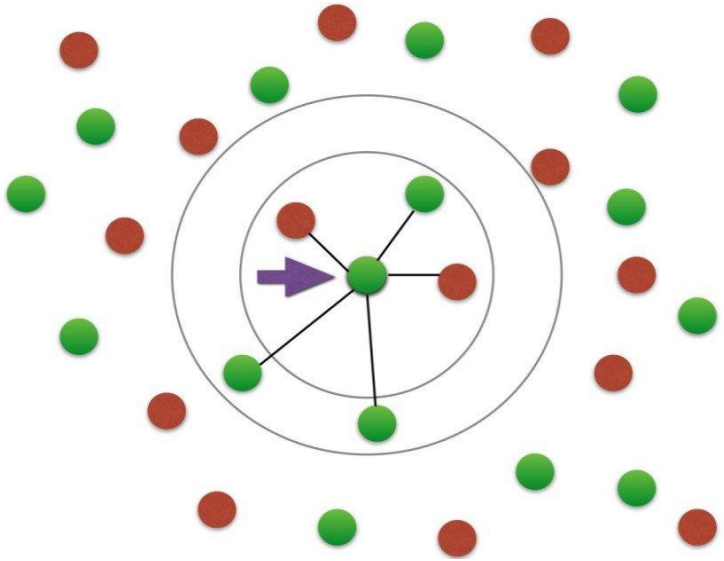
В частности для задачи классификации данный принцип можно переформулировать:

“Скажи мне, какой класс у твоих ближайших соседей, и я скажу какой класс у тебя”



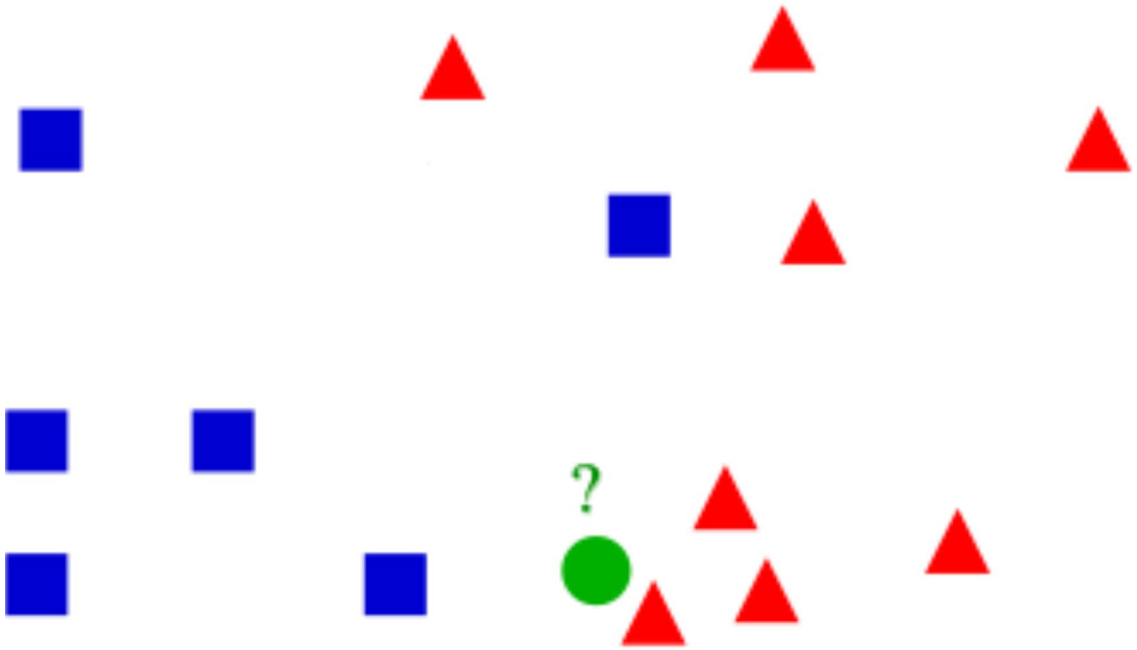
Идея работы

В основе работы классификатора, лежит иде
о том, что если объекты находятся близко
друг к другу в некотором метрическом
пространстве, то и их метки также близки.



Метод ближайших соседей

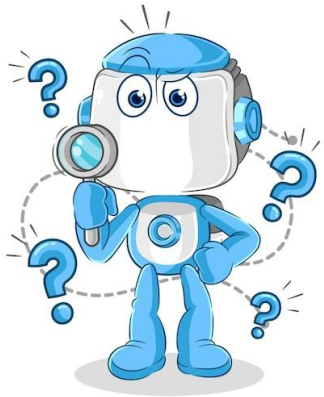
Как классифицировать новый объект?



Метод ближайших соседей

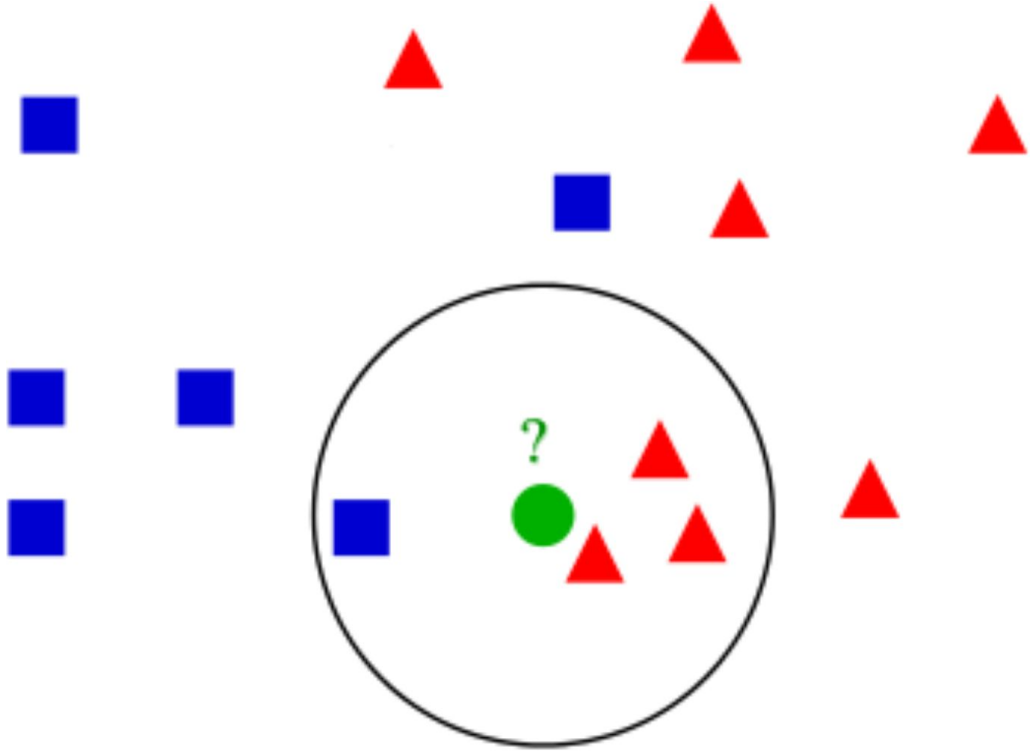
Чтобы классифицировать новый объект, нужно:

- Вычислить расстояние до каждого из объектов обучающей выборки
- Выбрать k объектов обучающей выборки, расстояние до которых минимально
- Класс классифицируемого объекта – это класс, наиболее часто встречающийся среди k ближайших соседей



Метод ближайших соседей

Число ближайших соседей
k – гиперпараметр метода.
Например, для $k = 4$.
Объект будет отнесён к
классу “треугольник”



Формализация метода

Пусть k – количество соседей. Для каждого объекта u возьмём k ближайших к нему объектов из тренировочной выборки:

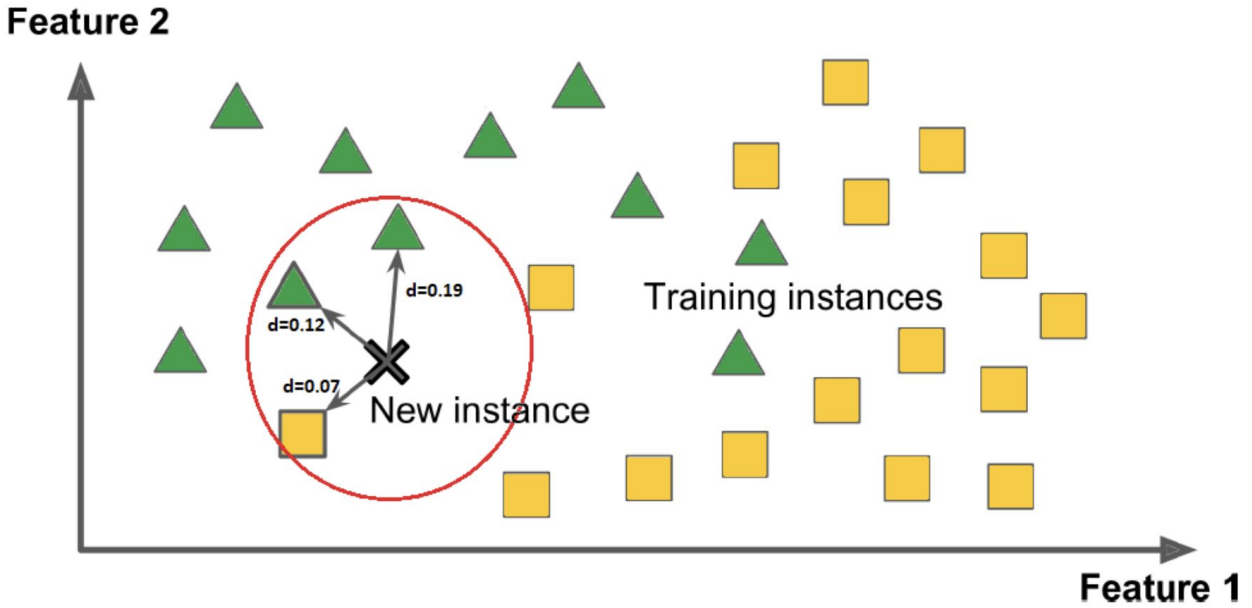
$$x_{(1;u)}, x_{(2;u)}, \cdots, x_{(k;u)}$$

Тогда класс объекта u определяется следующим образом:

$$a(u) = argmax_{y \in Y} \sum_{i=1}^{k} I[y(x_{(i,u)}) = y]$$

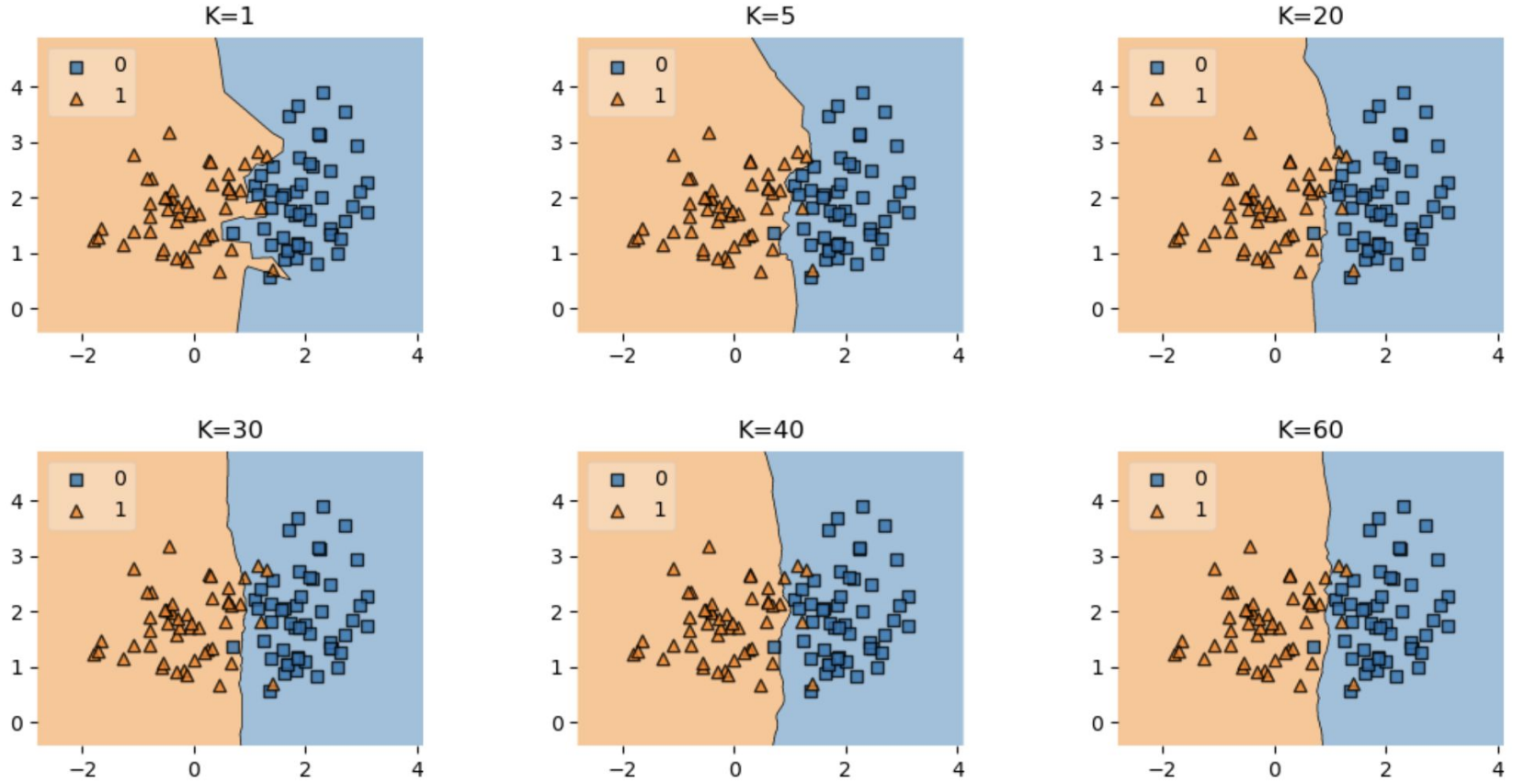
Взвешенный K-NN

Не все соседи равноценны по своему вкладу в выбор класса объекта, тк разное расстояние до классифицируемого объекта. Поэтому тем объектам, которые лежат ближе припишем больший вес.

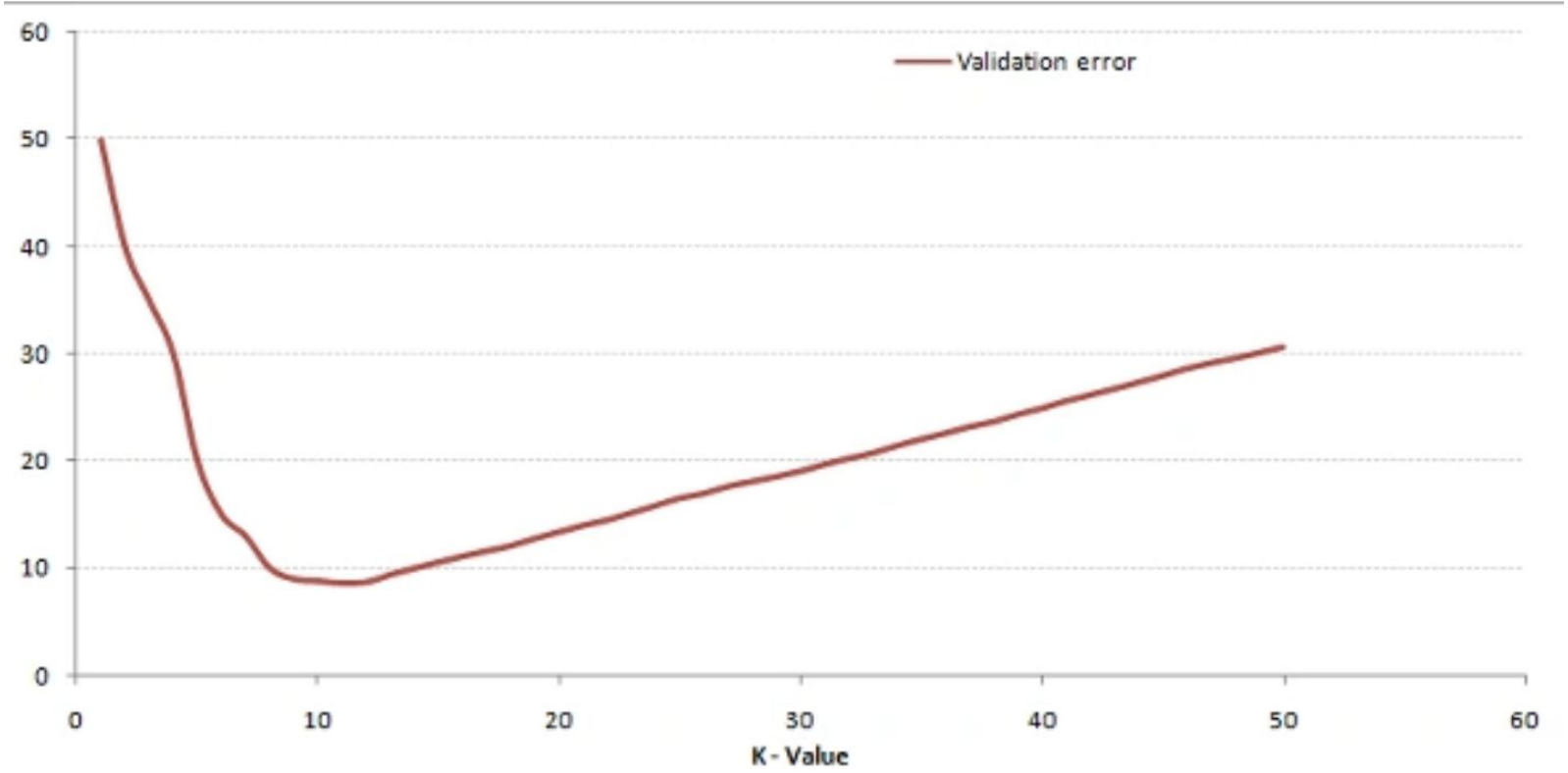


$$a(u) = argmax_{y \in Y} \sum_{i=1}^{k} \frac{1}{\rho(u, x_{(i,u)})} I[y(x_{(i;u)}) = y]$$

Влияние гиперпараметра k



Оптимальное k



Как измеряем расстояние

Поиск ближайших соседей в признаковом пространстве производится на основе некоторой, метрики ϱ:

$$\rho : R^d \times R^d \to R$$

Примерами такой метрики может служить

- Евклидова метрика
- Манхэттенское расстояние
- Расстояние Хэмминга

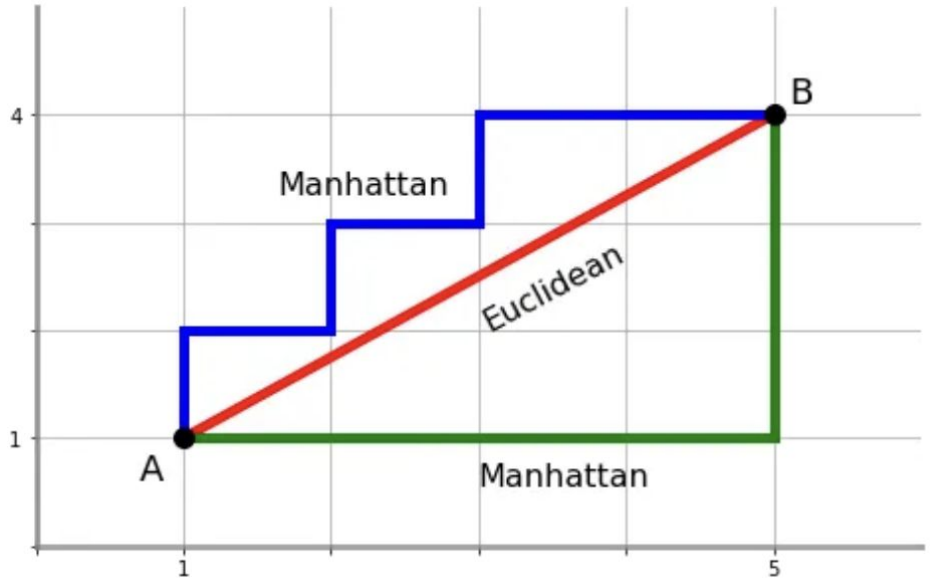
Манхэттенское расстояние

Евклидова метрика:

$$\rho(A, B) = \sqrt{\sum_{i=1}^{d} (a_i - b_i)^2}$$

Манхэттенское расстояние:

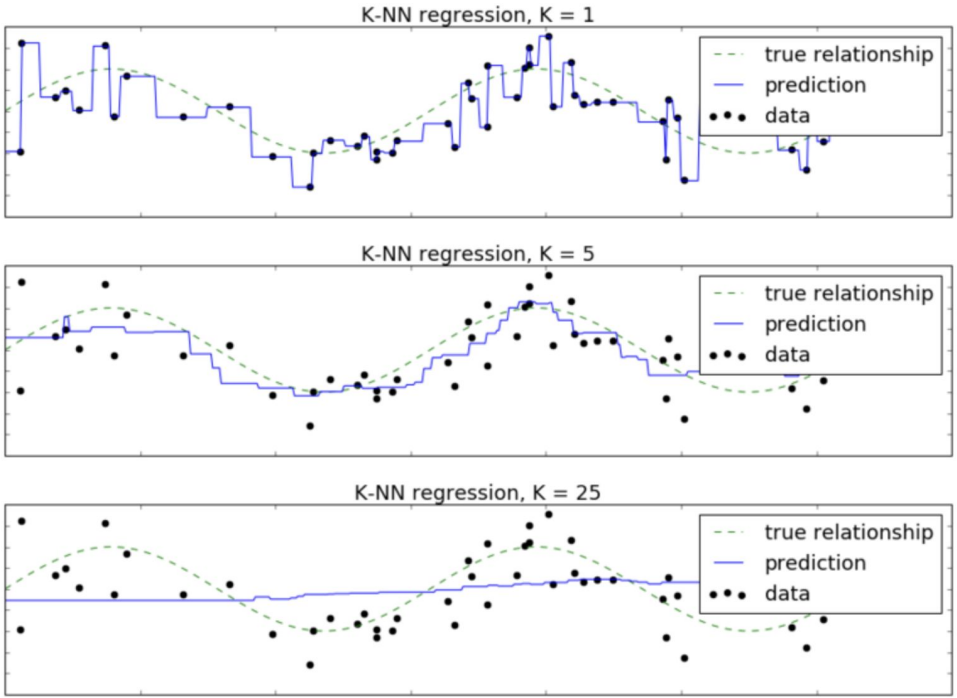
$$\rho(A, B) = \sum_{i=1}^{d} |a_i - b_i|$$



Вопрос: Если в линейных моделях на стадии обучения происходит поиск оптимальных весов w, в наивном байесе вычисление статистик, то что в методе ближайших соседей?

K-NN для задачи регрессии

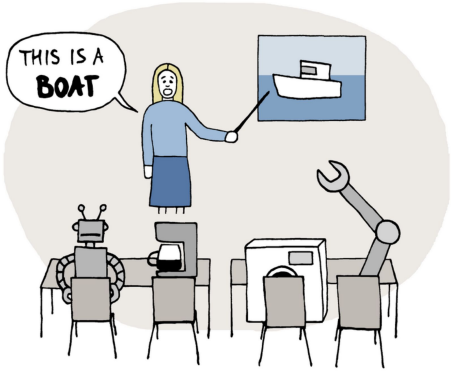
$$a(u) = \frac{1}{k} \sum_{i=1}^{k} y(x_{(i;u)})$$



Особенности применения

- Необходимо иметь достаточно памяти, чтобы хранить все объекты обучающей выборки
- В случае больших выборок алгоритм, может долго работать, так как для данного объекта необходимо вычислить расстояние до **всех** объектов
- Перед использованием необходимо **масштабировать** данные, иначе признаки с большими числовыми значениями будут доминировать при вычислении расстояний

Отсев признаков



Зачем нужен отбор признаков?

Не всегда наличие множества признаков в модели приводят к её улучшениям. В некоторых случаях модель может иметь бесполезные признаки, которые:

- Могут коррелировать друг с другом – проблема мультиколлинеарности в линейных моделях
- Могут значительно увеличивать расчётное время предсказания
- Нести много шумных значений в своих значениях

Отсев по дисперсии

- Удаляем те признаки, которые имеет маленькую дисперсию – меньшую некоторого порога T, так как данные признаки близки к константам

$$DX_i = E(X_i - EX_i)^2 < T$$

Отбор по корреляции с таргетом

- Для каждого признака вычислим его корреляцию с целевой переменной. Будем выкидывать признаки, имеющие маленькую корреляцию

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

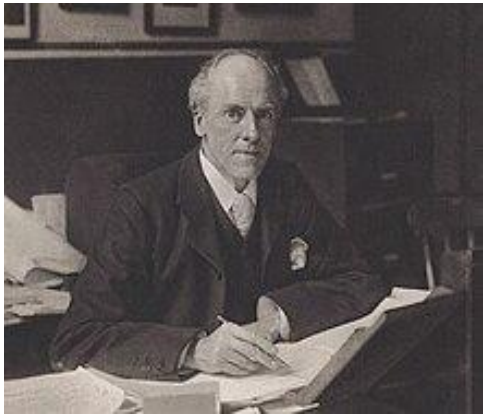
Более сложные методы

- Фильтрационные методы (Filtration methods)
- Обёрточные методы (Wrapping methods)
- Встроенные в модель отбор (Model selection)

Фильтрационные методы

- Фильтрационные методы – это отбор признаков по различным статистическим тестам.

  Идея метода состоит в вычислении влияния каждого признака в отдельности на целевую переменную с помощью вычисления некоторой статистики.



Карл Пирсон

Технический инструментарий

В Sklearn есть сразу несколько методов, использующих отбор по статистическим критериям.

Среди них выделим следующие:

- **SelectKBest** – оставляет *k* признаков с наибольшим значением статистики
- **SelectPercentile** – оставляет признаки со значениями выбранной статистики, попавшими в заданную пользователем квантиль

Статистические тесты для отбора признаков

- Тест χ^2 – используется в статистике для проверки независимости двух событий.
- Поскольку χ^2 проверяет степень независимости между двумя переменными, а мы хотим сохранить только признаки, наиболее зависимые от метки, то будем вычислять χ^2 между каждым признаком и меткой, сохраняя признаки с наибольшими значениями
- Критерий χ^2 может применяться только для бинарных или порядковых признаков

Формула вычисления

- Статистика χ^2 вычисляется по формуле:

$$\chi^2 = \sum_{i=1}^{N} \sum_{j=1}^{M} \frac{O_{ij} - E_{ij}}{E_{ij}}$$

Где O_{ij} – наблюдаемая частота, E_{ij} – ожидаемая частота.

- Наблюдаемая частота в точности равна значениям в таблице
- Ожидаемая частота есть – математическое ожидание некоторого события A

Пример

Пример: Хотим выявить влияние курения на гипертонию

	Артериальная гипертония есть (1)	Артериальной гипертонии нет (0)	**Всего**
Курящие (1)	40	30	**70**
Некурящие (0)	32	48	**80**
Всего	**72**	**78**	**150**

$$P(\text{курит, гипертония}) = P(\text{курит}) \cdot P(\text{гипертония}) = \frac{70}{150} \cdot \frac{72}{150} = \frac{5040}{22500}$$

$$E_{0,0} = \sum_{i=1}^{150} P(\text{курит, гипертония}) = 150 * \frac{5040}{22500} = 33.6$$

Пример

Вычислим подобным образом все ожидаемые наблюдения

	Артериальная гипертония есть (1)	Артериальной гипертонии нет (0)	**Всего**
Курящие (1)	(70*72)/150 = 33.6	(70*78)/150 = 36.4	**70**
Некурящие (0)	(80*72)/150 = 38.4	(80*78)/150 = 41.6	**80**
Всего	**72**	**78**	**150**

Посчитаем полную статистику

$$\chi^2 = \frac{(40-33.6)^2}{33.6} + \frac{(30-36.4)^2}{36.4} + \frac{(32-38.4)^2}{38.4} + \frac{(48-41.6)^2}{41.6} = 4.396$$

При отборе оставляем k признаков с наибольшей статистической значимостью

Другие статистики значимости

- Mutual Information

  Для векторов X и Y статистика вычисляется по формуле

$$I(X,Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \cdot log \frac{p(x,y)}{p(x) \cdot p(y)}$$

Обёрточные методы

Обёрточные методы используют жадный отбор признаков, т.е. последовательно выкидывают наименее подходящие по мнению методов признаки.

В Sklearn есть обёрточный метод – Recursive Feature Elimination (RFE).

Параметры метода:

- алгоритм, используемый для отбора признаков (например, Random Forest)
- число признаков, которое хотим оставить

Обёрточные методы

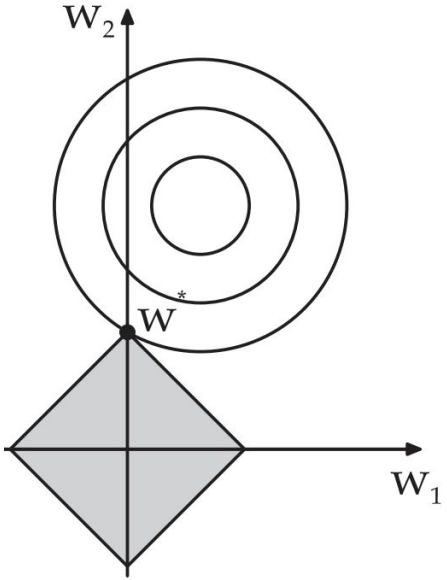
- Шаг-1: Перебираем все признаки и убираем тот, удаление которого сильнее всего уменьшает ошибку
- Шаг-2: Из оставшихся признаков убираем тот, удаление которого сильнее всего уменьшает ошибку

  Повторяем шаги 1 и 2

Встроенные в модель методы

Напоминание: L_1– регуляризация умеет отбирать признаки

$$Q(a, X) + \alpha \sum_{i=1}^{d} |w_i| \to min_w$$



Информационные критерии

- **Информационный критерий** – мера качества модели, учитывающий степень степень “подгонки” модели под данные с корректировкой (штрафом) на используемое количество параметров.
- Информационные критерии основаны на **компромиссе между точностью и сложностью модели**. Критерии различаются тем, как они обеспечивают этот баланс

Критерий AIC

Критерий Акаике (AIC, Akaike Information Criterion) для линейных моделей:

$$AIC(a, X) = Q(a, X) + \frac{2\hat{\sigma}^2}{n} \cdot l$$

- Q — функционал ошибки
- $\hat{\sigma}$ — оценка дисперсии ошибки $D(y_i - a(x_i))$
- l — количество используемых признаков
- n — число объектов

Отбор с помощью информационных критериев

- Если в модели k признаков, то существует 2^k всевозможных моделей
- В идеале необходимо построить все 2^k моделей, для каждой посчитать значение критерий качества (AIC) и выбрать модель, лучшую по этому критерию
- При большом количестве регрессоров используют метод включений-исключений жадным образом

Пример

Задача предсказания уровня преступности в разных штатах по следующим признакам:

Регрессор
Нулевой коэффициент
Возраст
Южный штат(да/нет)
Образование
Расходы
Труд
Количество мужчин
Численность населения
Безработные (14-24)
Безработные (25-39)
Доход

Пример

В модели с полным набором регрессоров AIC = -310.37. В порядке убывания AIC при удалении каждой из переменных равен:

Численность населения (AIC = -308), Труд (AIC = -309), Южный штат (AIC = -309), Доход (AIC = -309), Количество мужчин (AIC = -310), Безработные I (AIC = -310), Образование (AIC = -312), Безработные II (AIC = -314), Возраст (AIC = -315), Расходы (AIC = -324).

Таким образом, имеет смысл удалить переменную “Население”.