

Как оценивать качество регрессионной модели?

Как оценивать качество регрессионной модели?

1. MSE
2. MAE
3. Корреляция Пирсона
4. Корреляция Спирмена
5. R^2 (по сути - та же корреляция Пирсона)

Как оценивать качество классификатора?

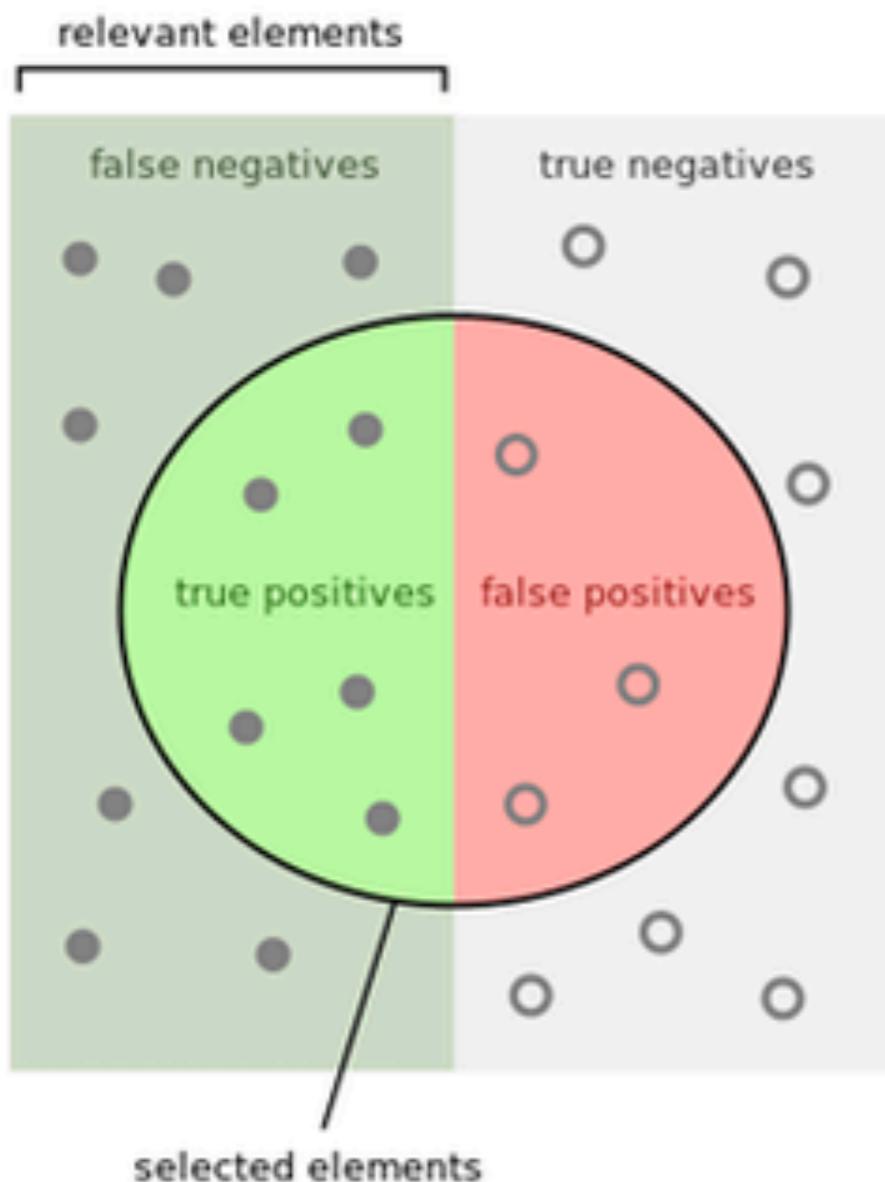
Как оценивать качество регрессионной модели?

- 1. Точность - сколько объектов предсказали
из выборки. Чем плохо?**

Как оценивать качество регрессионной модели?

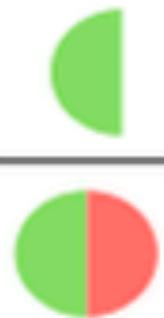
1. Точность - сколько объектов предсказали из выборки. Чем плохо? **Если модель предсказывает всегда 0, и у нас большая часть объектов принадлежит к первому классу - то 0 и будет.**

Как оценивать качество регрессионной модели?



How many selected items are relevant?

Precision =

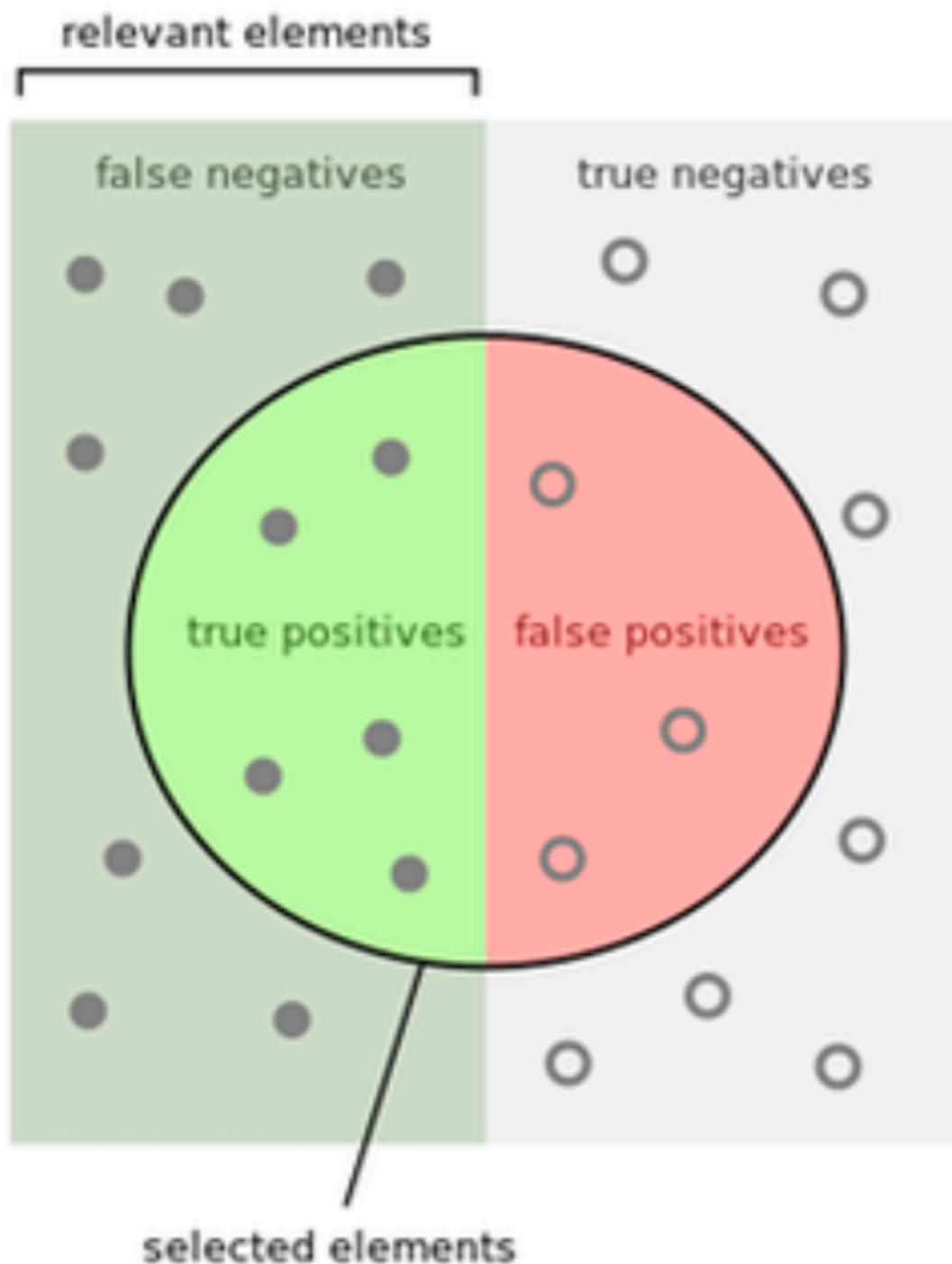


How many relevant items are selected?

Recall =



Как оценивать качество регрессионной модели?



$$specificity = \frac{TN}{TN + FP}$$

Какую долю объектов нулевого класса корректно определяем

F1-score

$$F1 - score = \frac{2 * (precision * recall)}{precision + recall}$$

Оцениваем качество модели, ранжирующей наши объекты по вероятности принадлежать 1 классу

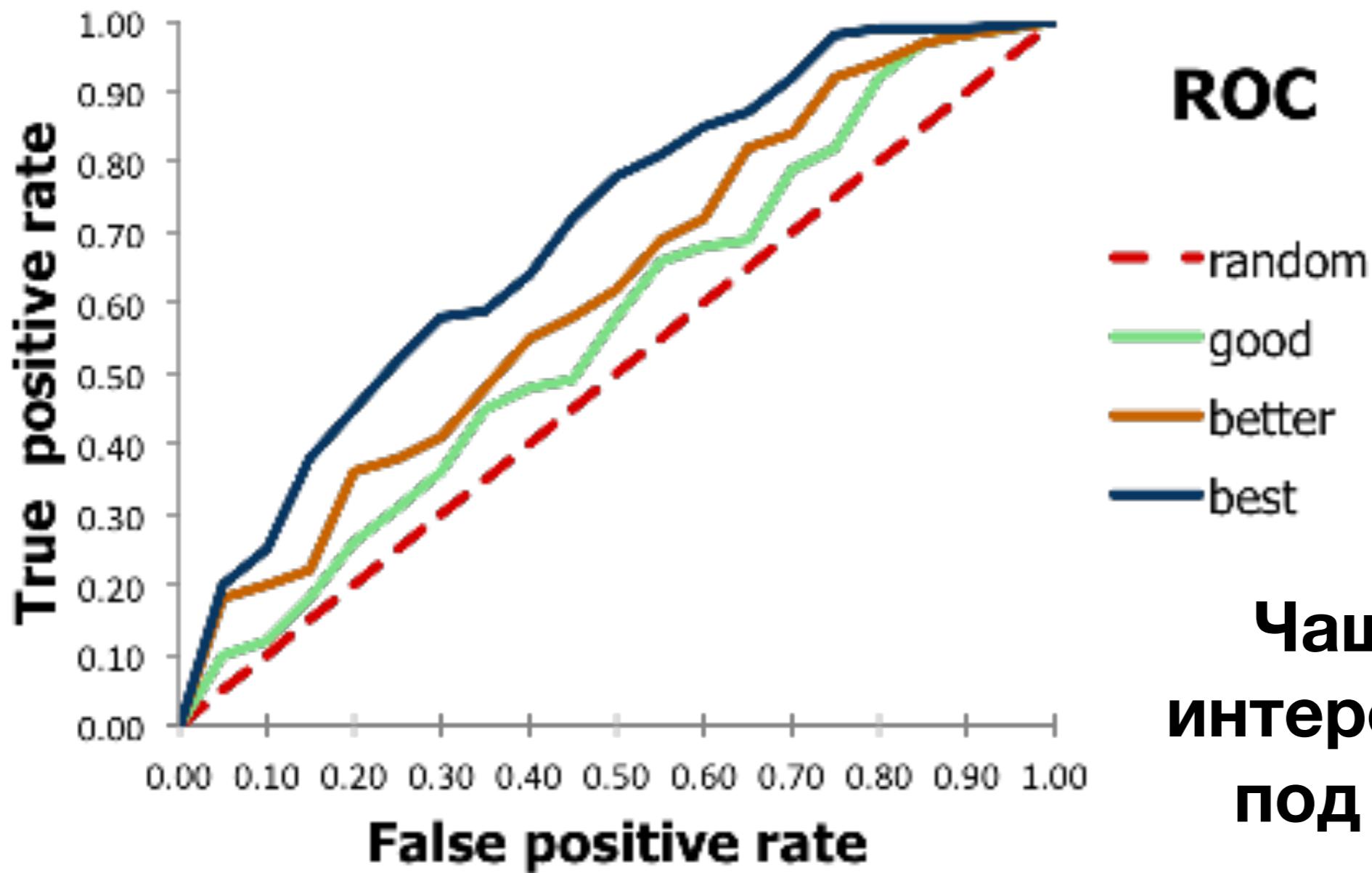
| Объект | Оценка от модели |
|--------|------------------|
| 1 | 2.56 |
| 2 | 2.03 |
| ... | .. |
| N | -0.79 |

Для каждого порога считаем **recall (sensitivity)** и **specificity**

$$recall = \frac{TP}{TP + FN} = \frac{TP}{P}$$

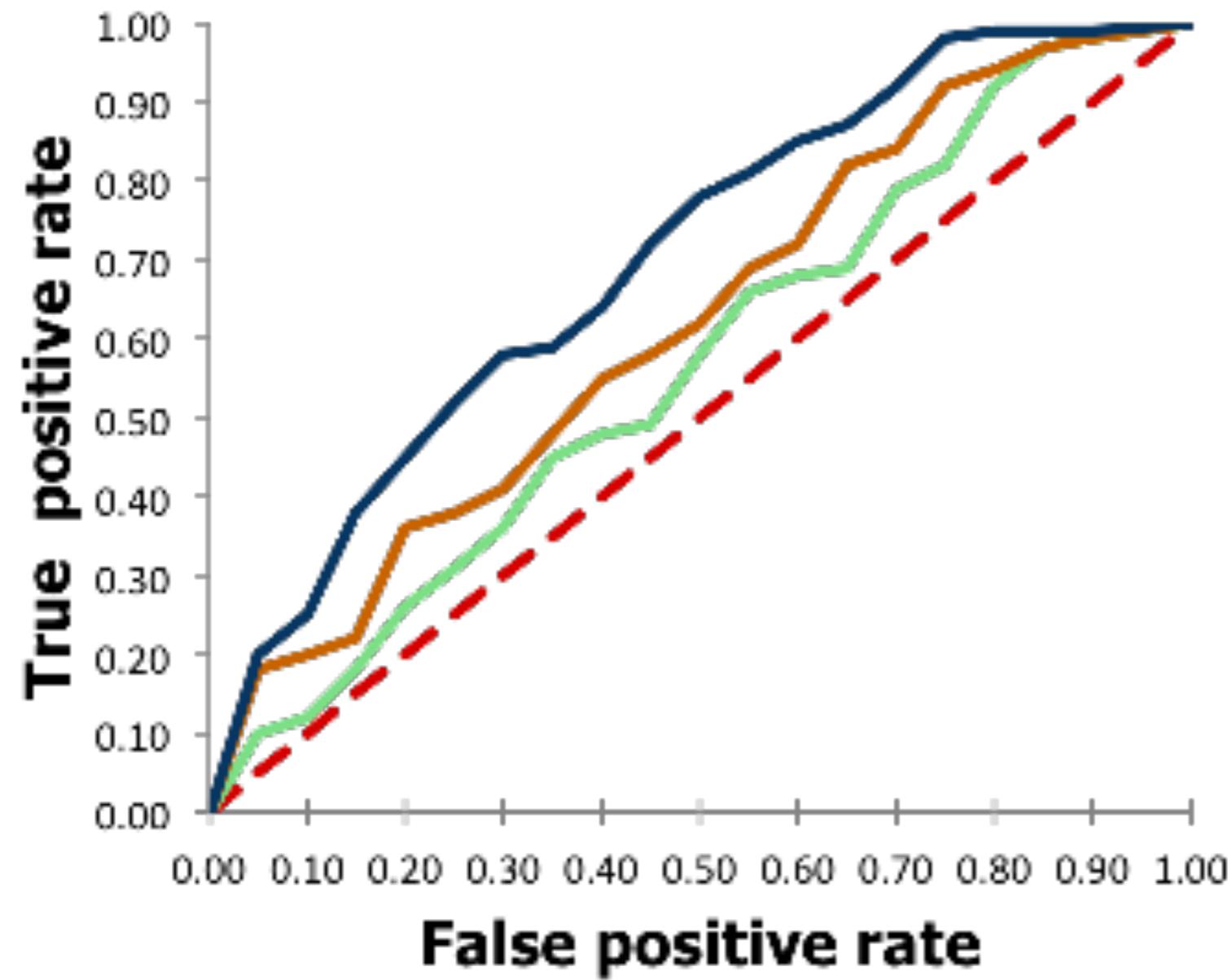
$$specificity = \frac{TN}{TN + FP} = \frac{TN}{N}$$

ROC-AUC



Чаще всего нас
интересует площадь
под ROC-кривой

ROC-AUC



ROC

- random
- good
- better
- best

Для случайного классификатора площадь - 0.5. Но - для случаев большого дисбаланса классов метрика может быть близка к 1 даже для плохого классификатора

Оцениваем качество модели, ранжирующей наши объекты по вероятности принадлежать 1 классу

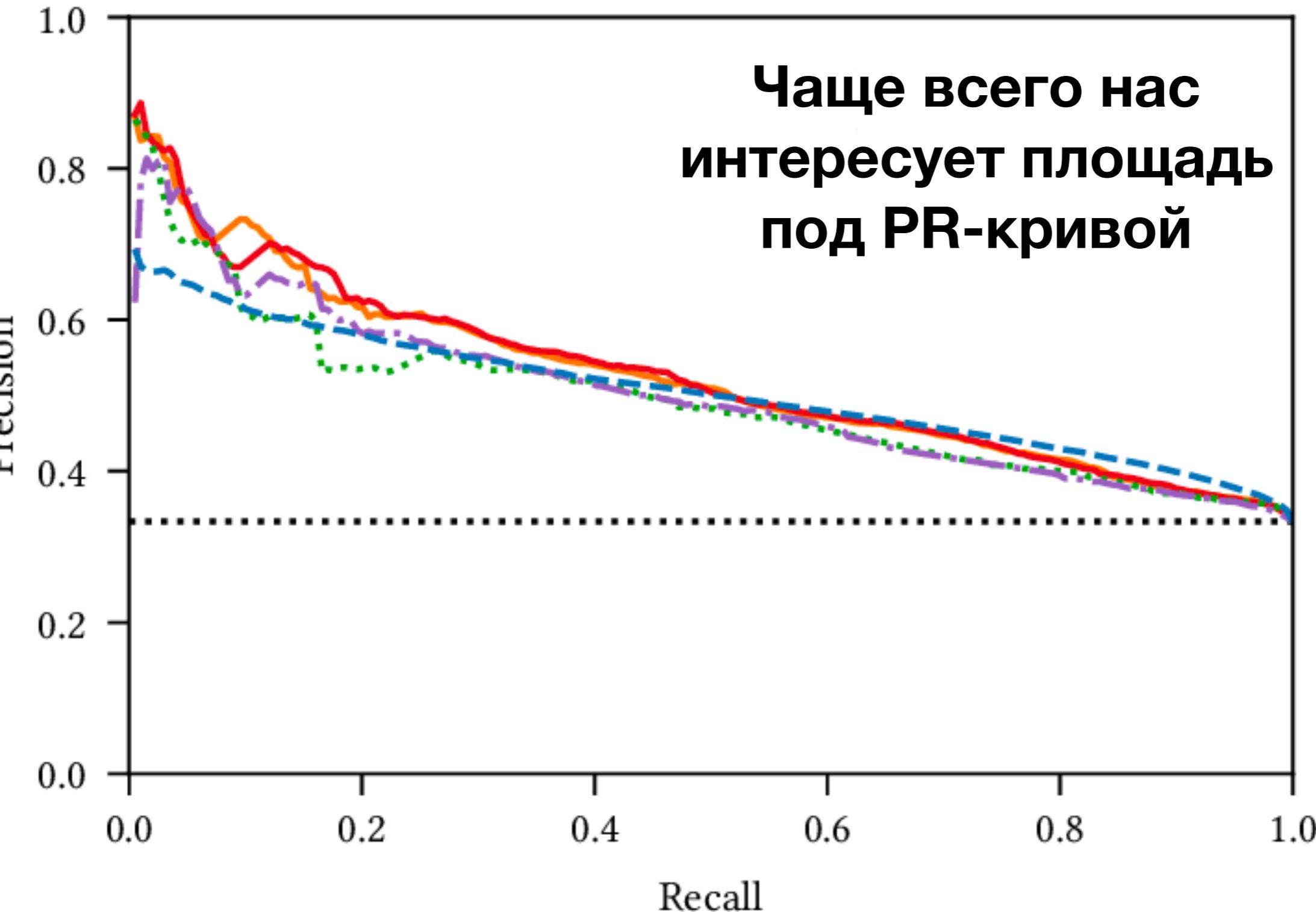
| Объект | Оценка от модели |
|--------|------------------|
| 1 | 2.56 |
| 2 | 2.03 |
| ... | .. |
| N | -0.79 |

Для каждого порога считаем **recall (sensitivity)** и **specificity**

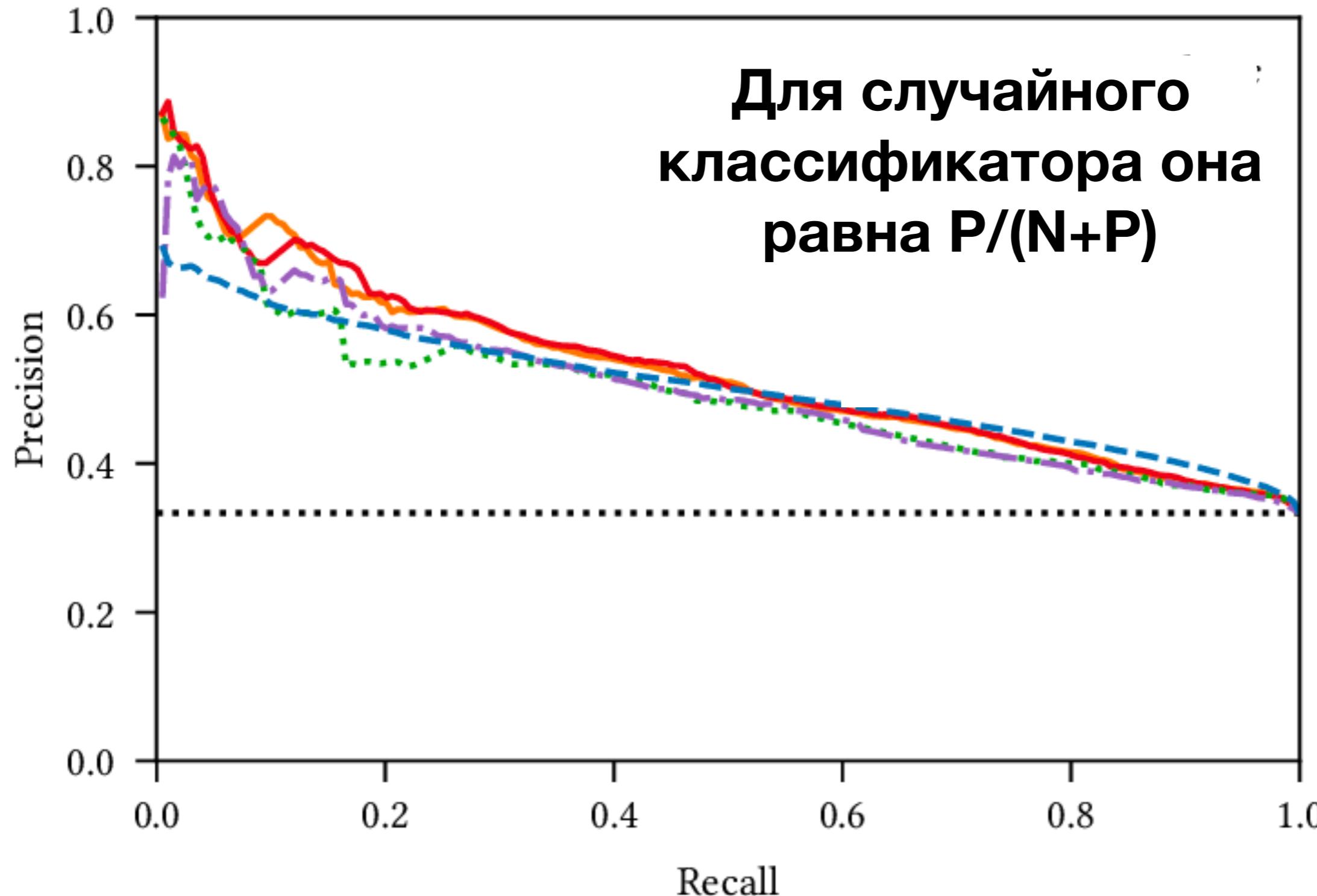
$$recall = \frac{TP}{TP + FN} = \frac{TP}{P}$$

$$precision = \frac{TP}{TN + FP}$$

PR-AUC



PR-AUC



Error correction codes

У нас есть сигнал

1110110011

В него вносятся ошибки:

1010110011

Что с этим делать?

Error correction codes

Передаем сигнал три раза, все три раза есть ошибки

```
1010110011  
1110110011  
1110110011
```

Восстанавливаем итоговый сигнал

```
1110110011
```

Машинное обучение

1111111111

10 объектов, все в реальности принадлежат классу 1

Пусть у нас есть три независимых классификатора, А, В и С. Каждый предсказывает 1 в 70% случаев

Ансамбли

Постановка задачи

Пусть дан набор объектов $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$, $\mathbf{x}_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, $i \in 1, \dots, N$, полученный из неизвестной закономерности $y = f(\mathbf{x})$. Необходимо построить такую $h(\mathbf{x})$, которая наиболее точно аппроксимирует $f(\mathbf{x})$.

Будем искать неизвестную

$$h(\mathbf{x}) = C(a_1(\mathbf{x}), \dots, a_T(\mathbf{x}))$$

$a_i(\mathbf{x}) : \mathcal{X} \rightarrow \mathcal{R}$, $\forall i \in \{1, \dots, T\}$ - базовые модели

$C : \mathcal{R} \rightarrow \mathcal{Y}$ - решающее правило

Простое голосование

$$h(\mathbf{x}) = \frac{1}{T} \sum_{i=1}^T a_i(\mathbf{x})$$



Машинное обучение

Все три классификатора верны: $0.7 * 0.7 * 0.7 = 0.3429$

Два классификатора верны: $0.7 * 0.7 * 0.3 + 0.7 * 0.3 * 0.7 + 0.3 * 0.7 * 0.7 = 0.4409$

Один верен: $0.3 * 0.3 * 0.7 + 0.3 * 0.7 * 0.3 + 0.7 * 0.3 * 0.3 = 0.189$

Все ошибаются: $0.3 * 0.3 * 0.3 = 0.027$

Если брать большинство голосов, то в 78% случаев мы правы

Три предсказания:

1111111100 = 80% accuracy
1111111100 = 80% accuracy
1011111100 = 70% accuracy.

Объединение дает качество 80%. Никакого увеличения, почему?

Корреляция

1111111100 = 80% accuracy

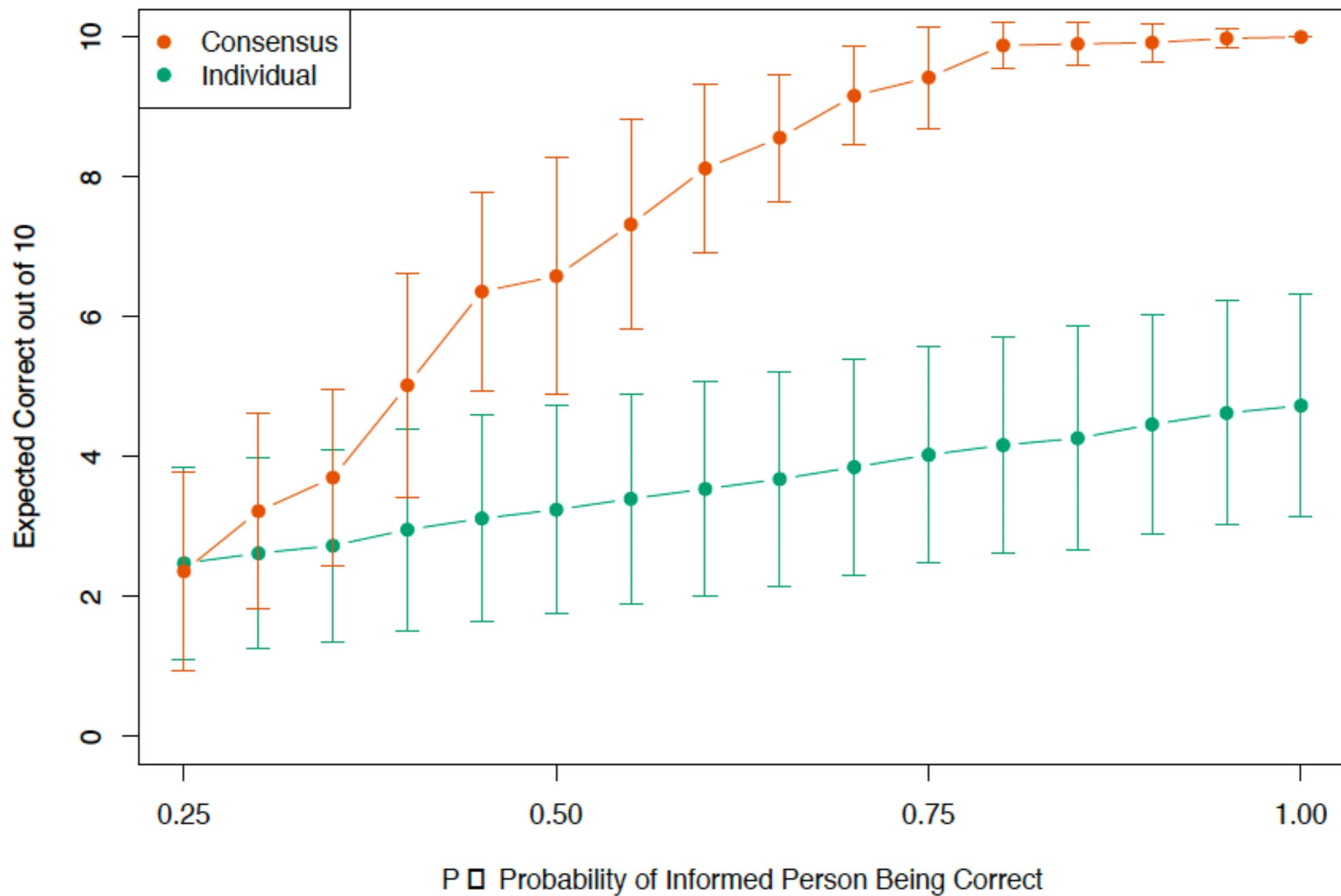
0111011101 = 70% accuracy

1000101111 = 60% accuracy

Получаем лучше предсказание!

1111111101 = 90% accuracy

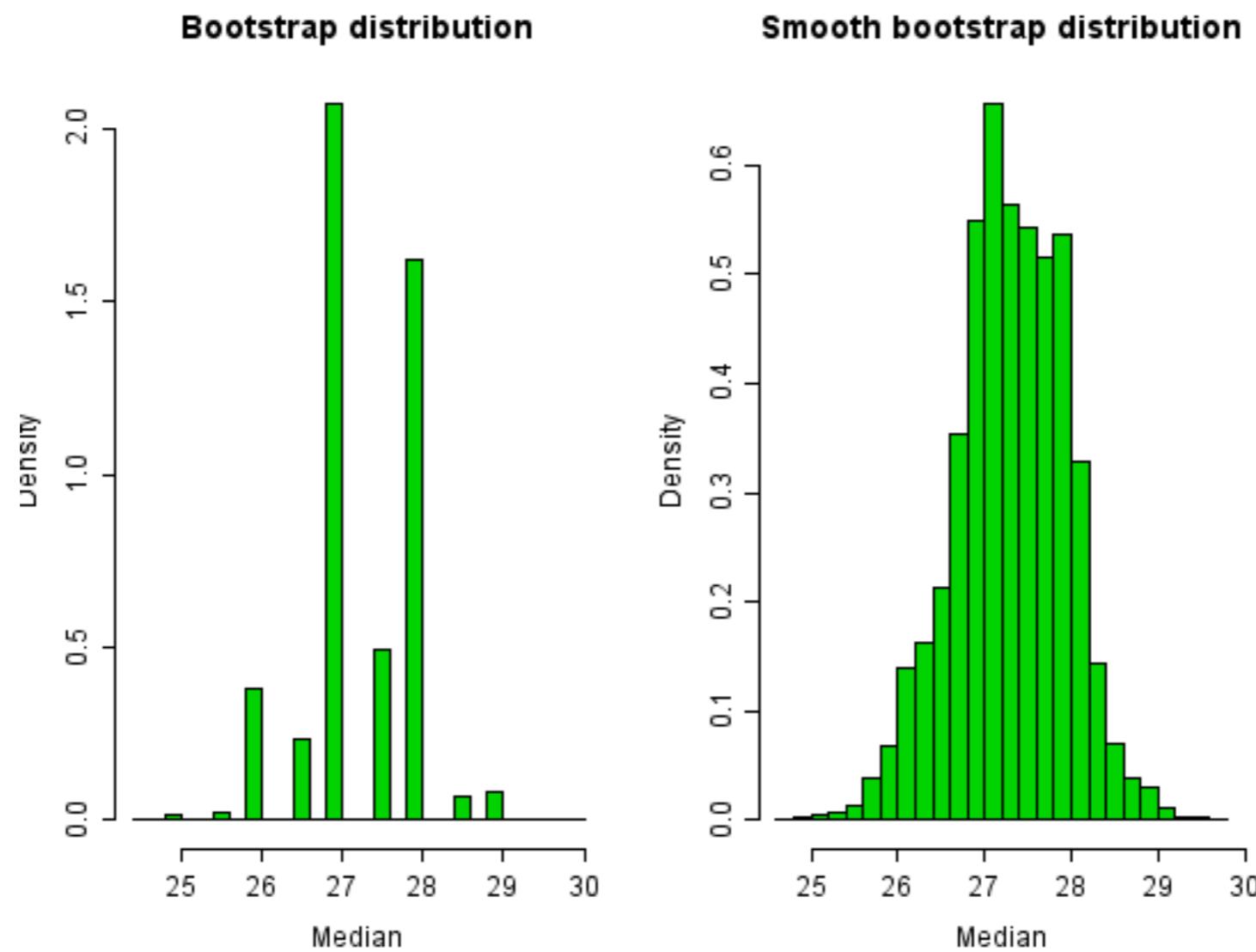
Wisdom of Crowds



Simulated academy awards voting. 50 members vote in 10 categories, each with 4 nominations. For any category, only 15 voters have some knowledge, represented by their probability of selecting the "correct" candidate in that category (so $P = 0.25$ means they have no knowledge). For each category, the 15 experts are chosen at random from the 50. Results show the expected correct (based on 50 simulations) for the consensus, as well as for the individuals. The error bars indicate one standard deviation. We see, for example, that if the 15 informed for a category have a 50% chance of selecting the correct candidate, the consensus doubles the expected performance of an individual.

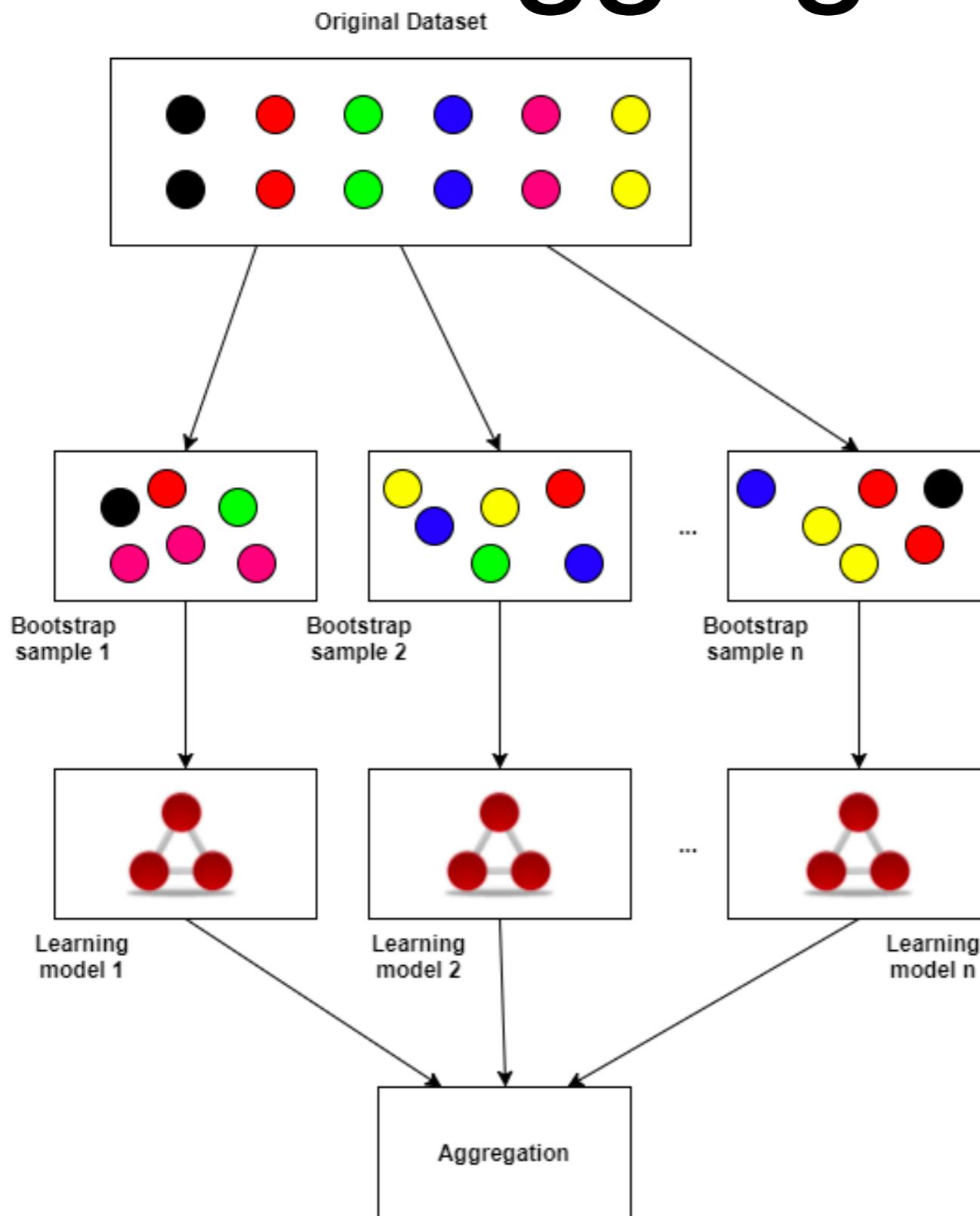
Bootstrap

Где уже встречали и зачем используется?



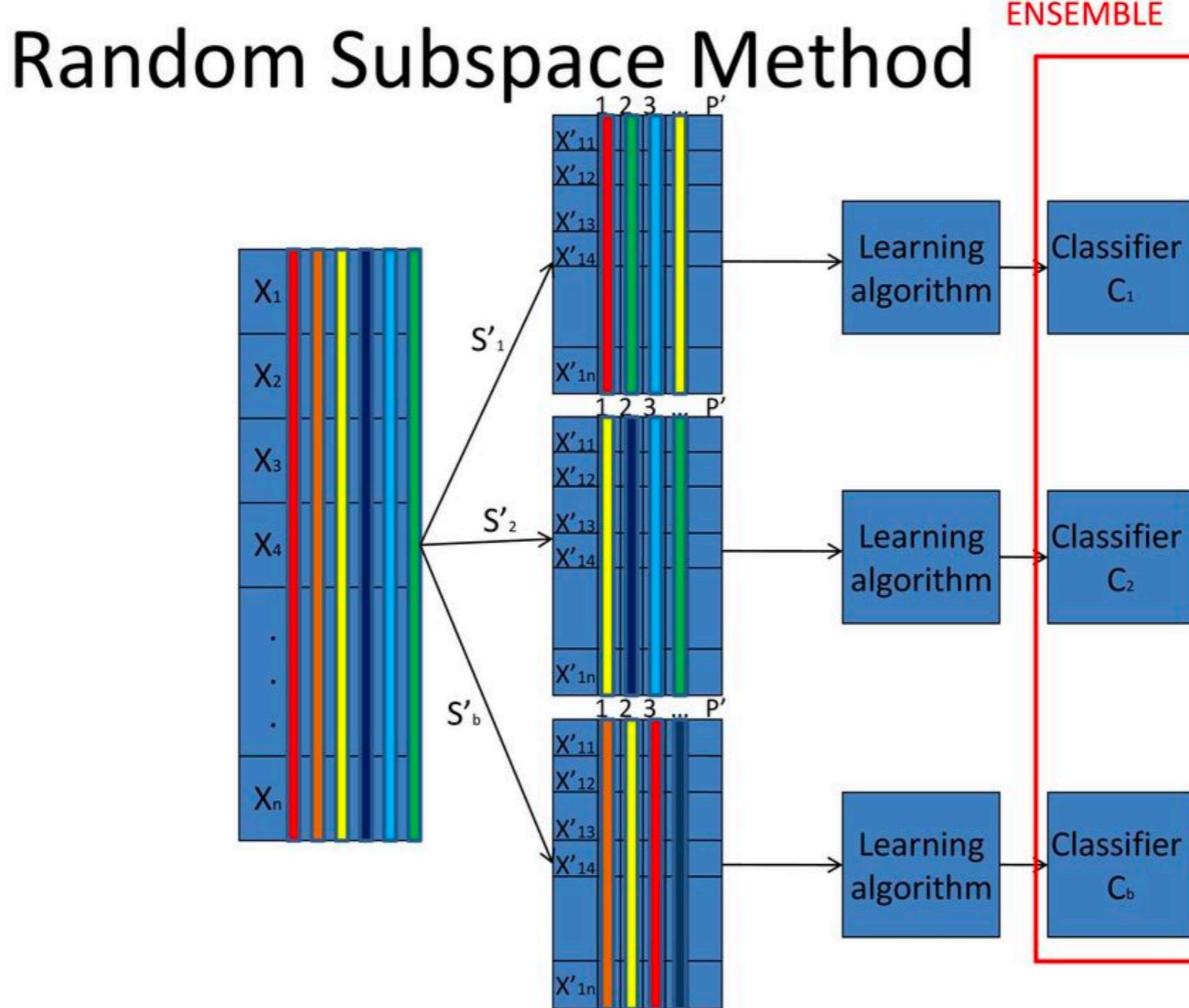
Делаем случайные выборки с повторениями такого же размера из нашего распределения. Считаем на каждой такой подборке некую величину - получаем распределение этой величины. Можем давать оценки значения, строить доверительные интервалы и тд

Bagging = Bootstrap aggregation

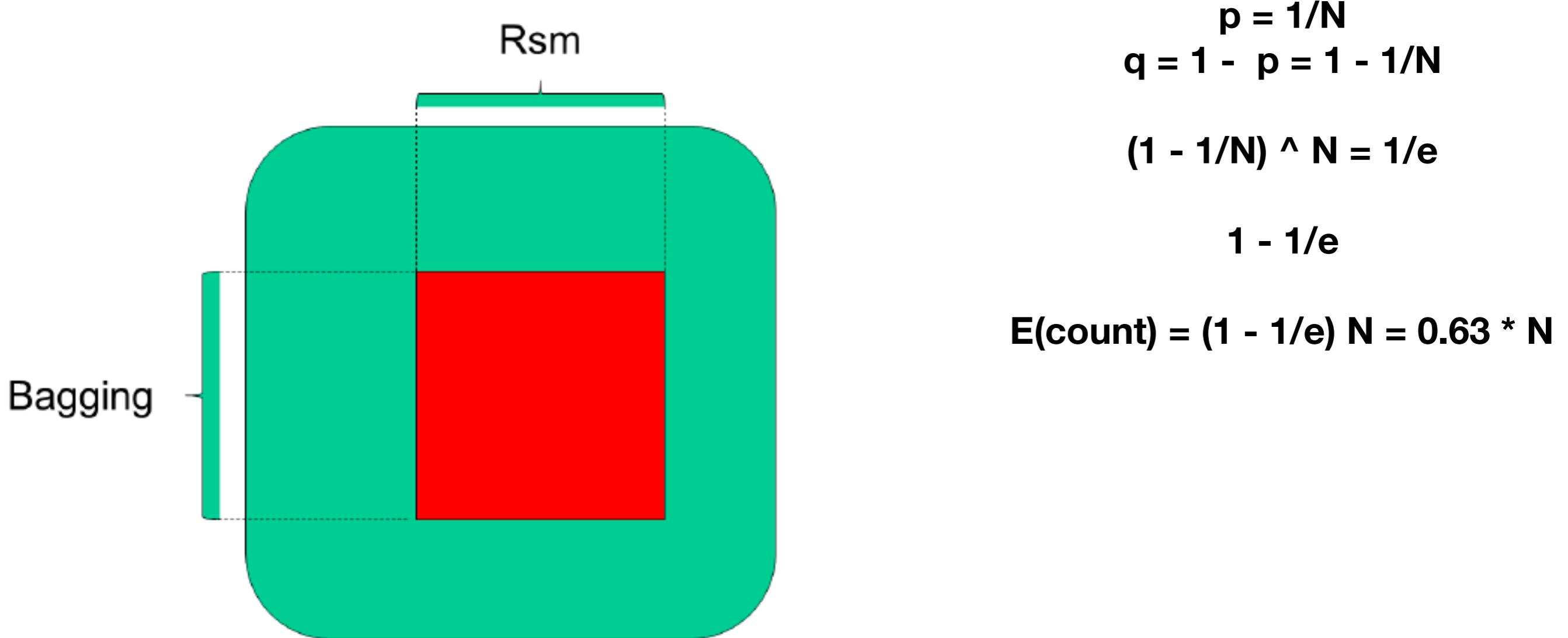


Обучаем алгоритмы по
случайным подборкам
размера N ,
полученным с
помощью выбора с
возвращением

RSM = random subspace method



Геометрическая интерпретация



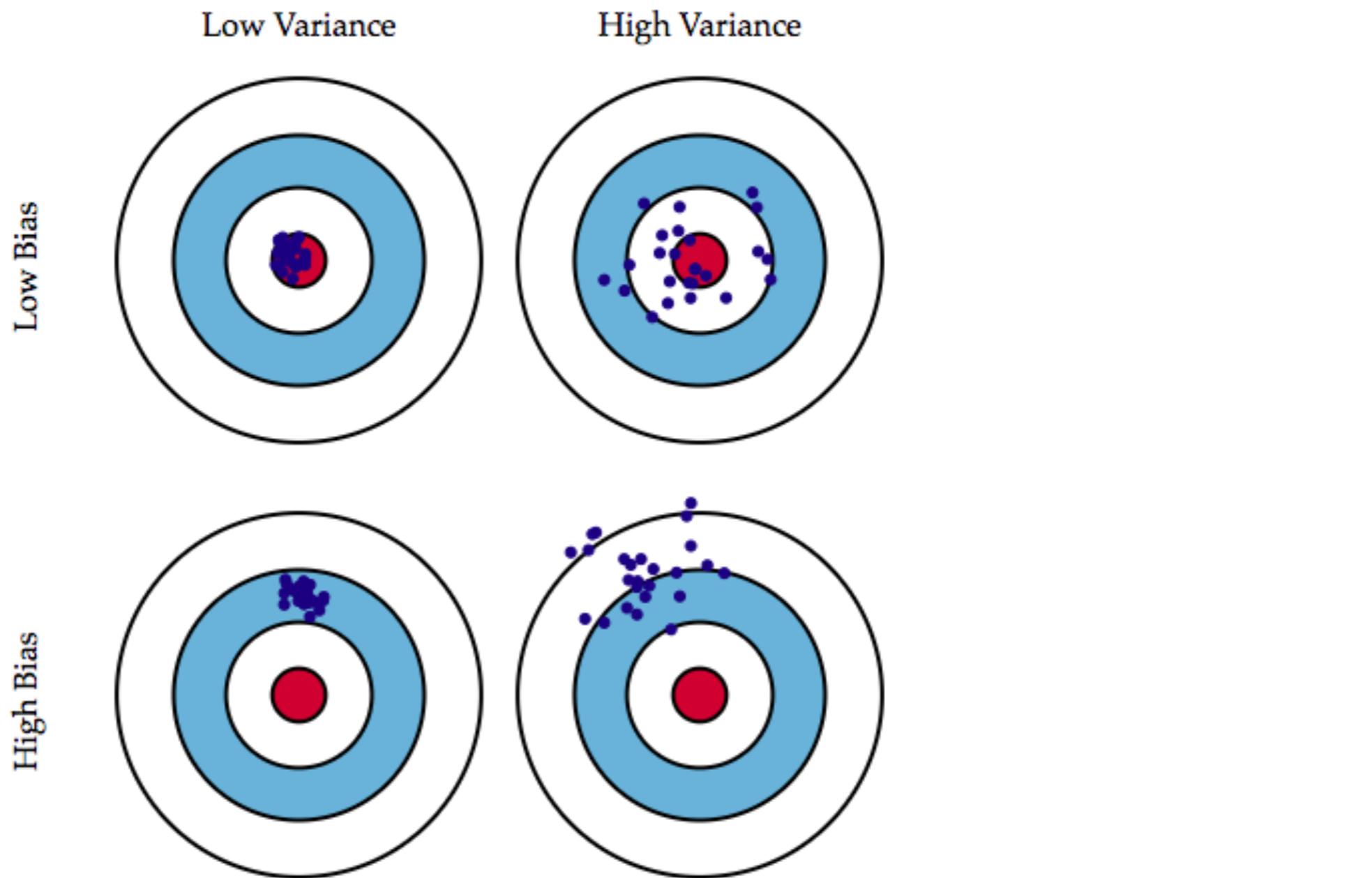
Bias-variance tradeoff

$$MSE = Var(y - h(x)) + (E(y - h(x)))^2 + Var(\epsilon)$$

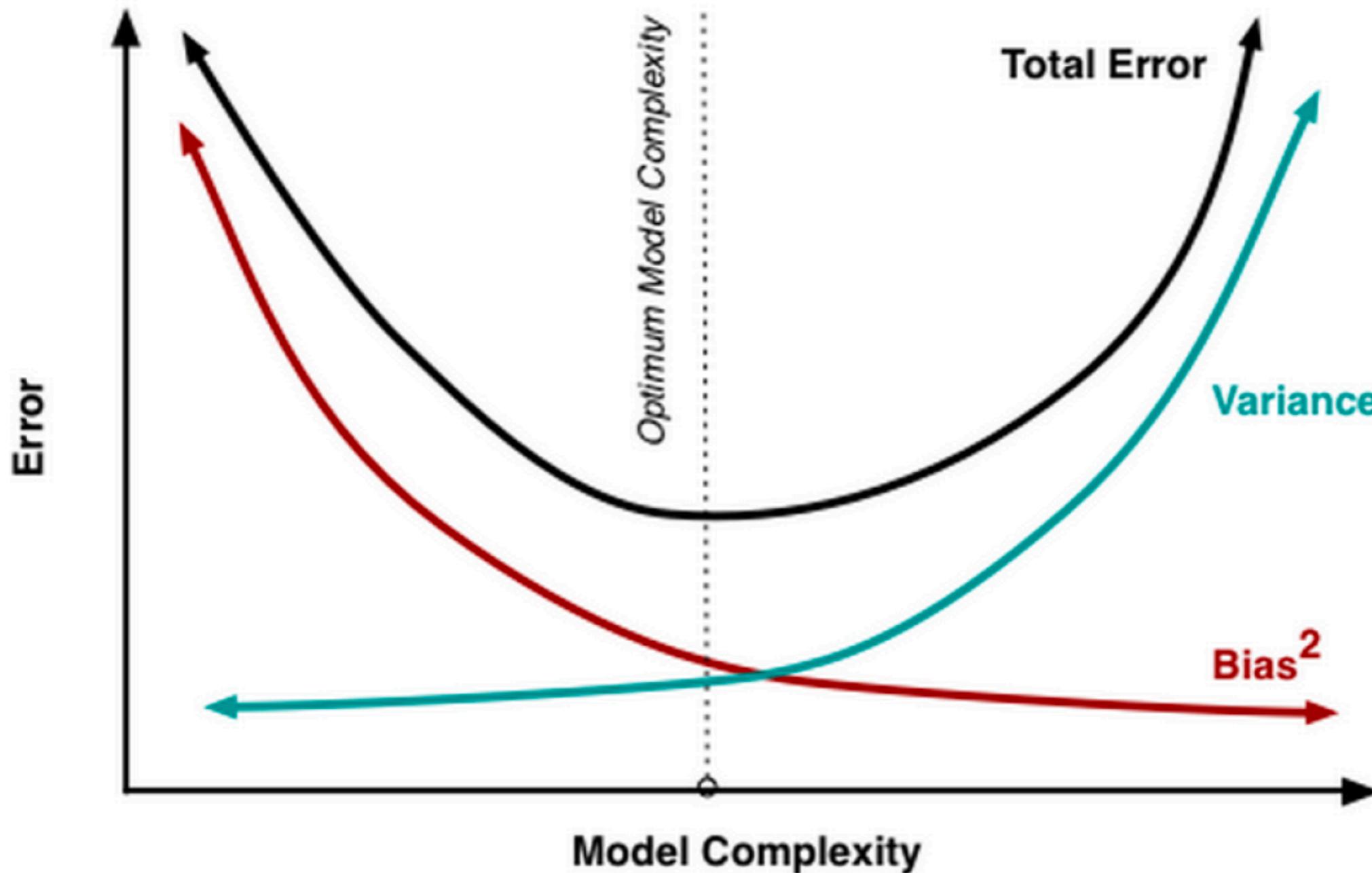
Variance модели

Bias модели

Шум в данных



Capacity



Простое голосование

$$Variance(ensemble) = \rho \cdot \sigma^2 + \frac{1 - \rho}{T} \sigma^2$$

ρ

- корреляция между двух моделями

σ^2

- Variance одной модели

T

- число баровых моделей

$$Bias(ensemble) = bias(base_model)$$

Деревья неустойчивы

- ▶ Незначительные изменения в данных приводят к значительным изменениям в топологии дерева



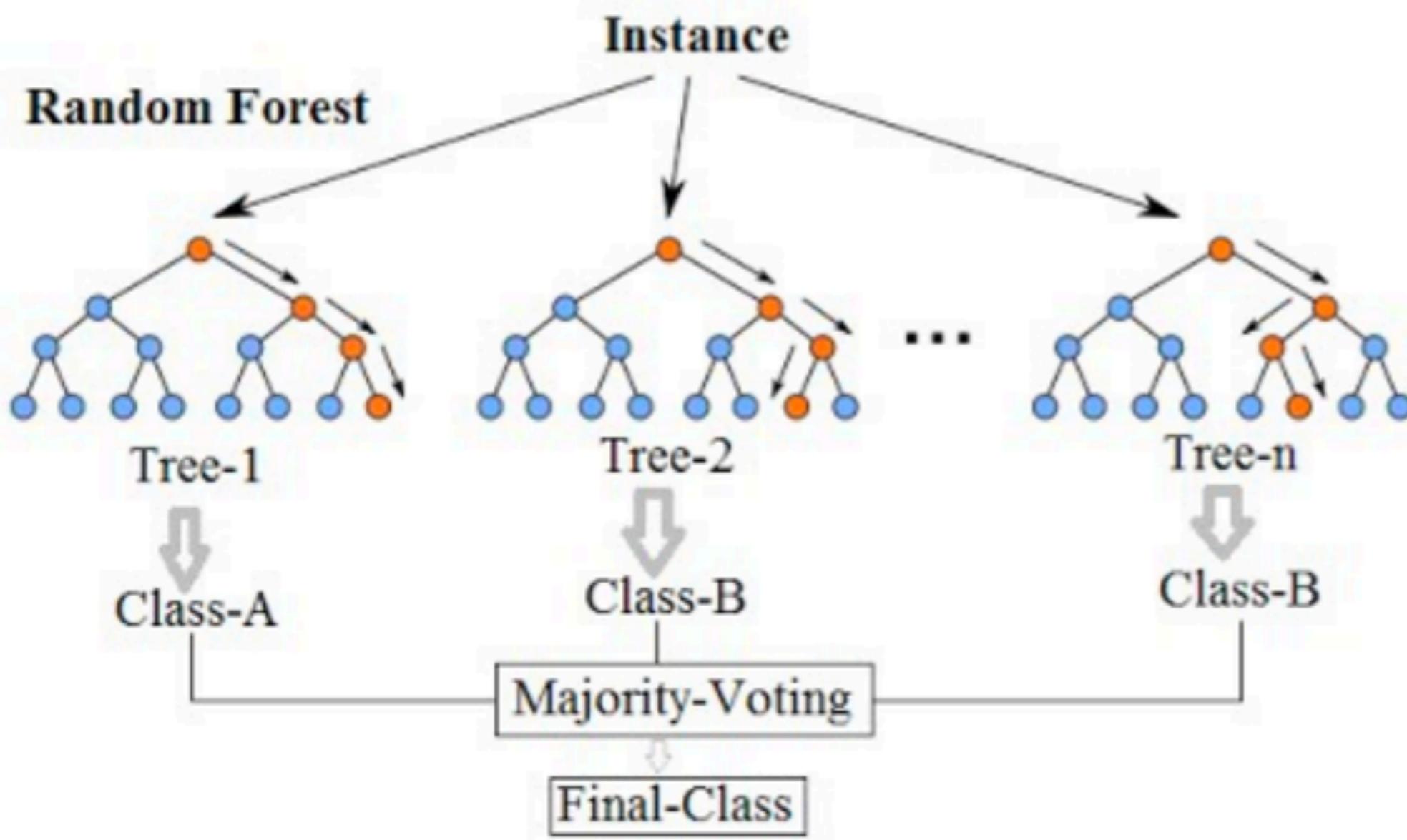
Взято из презентации Гулин В.,
Техносфера

Теорема об универсальном аппроксиматоре

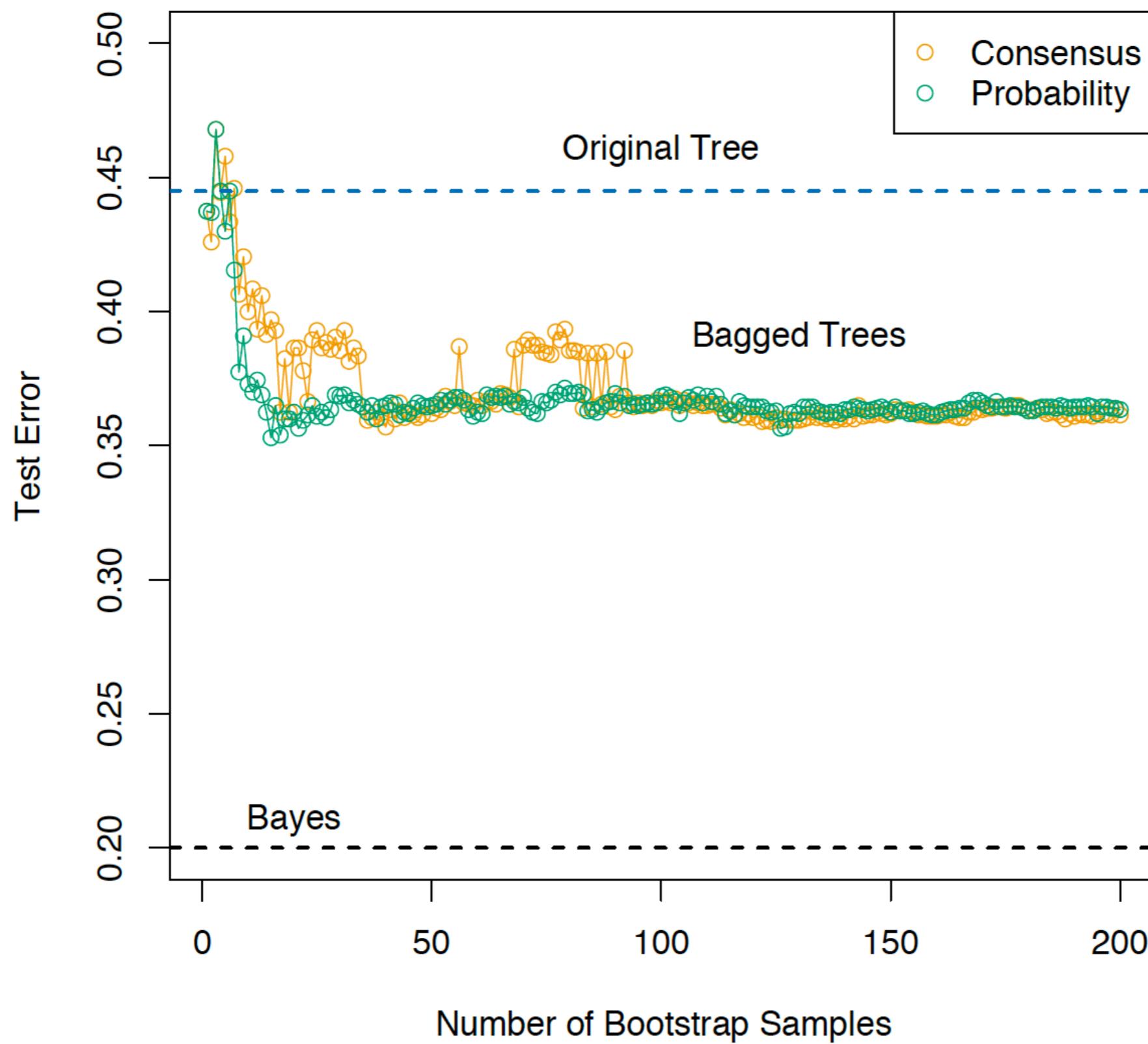
**С помощью дерева решений можно аппроксимировать любую
кусочно-заданную функцию**

Случайный лес

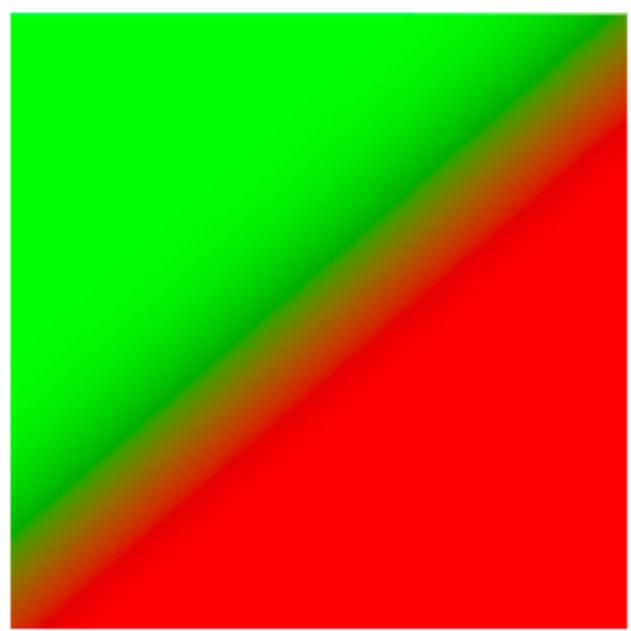
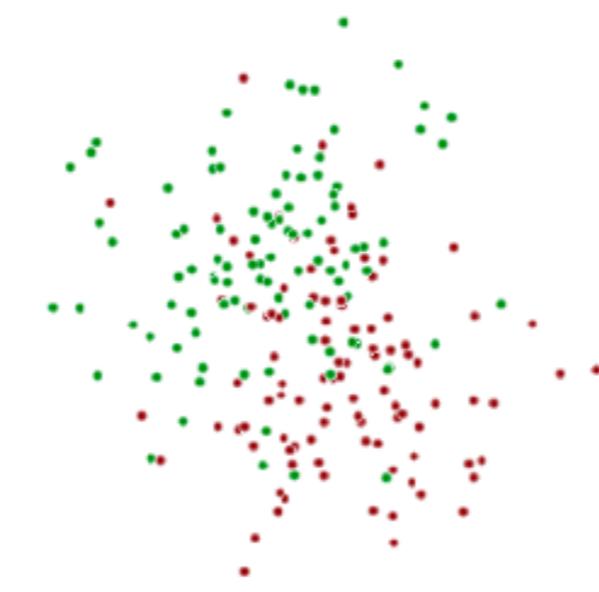
Random Forest Simplified



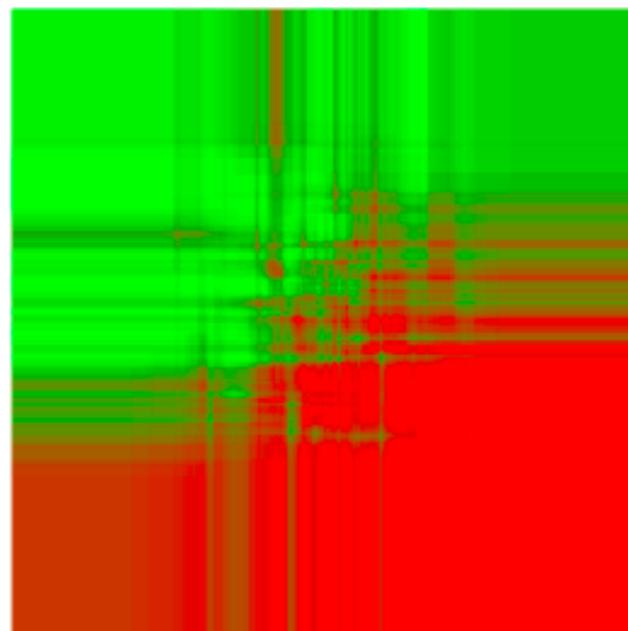
Случайный лес



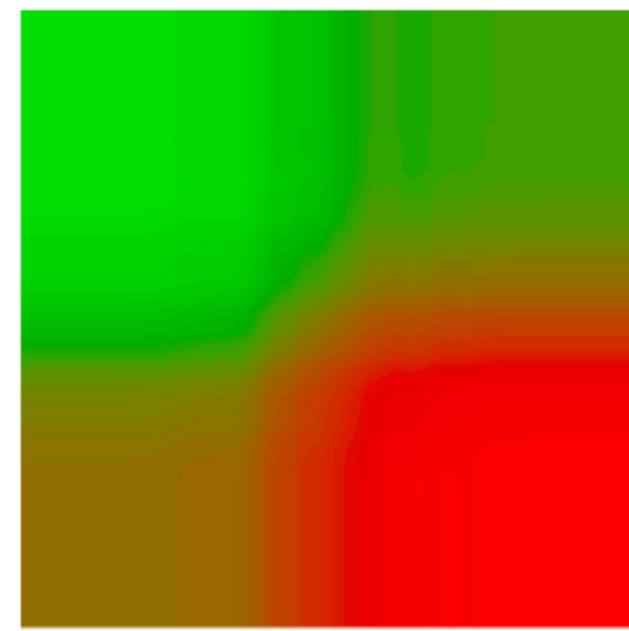
Случайный лес



(a) Original data



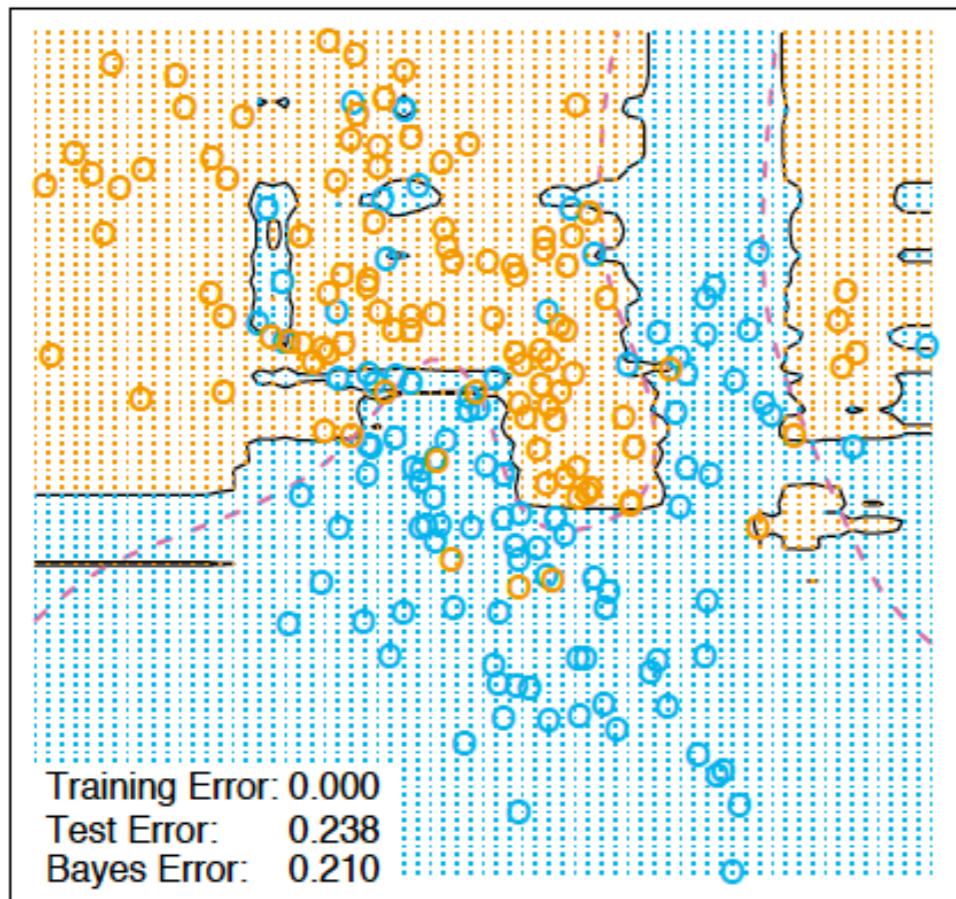
(b) RF (50 Trees)



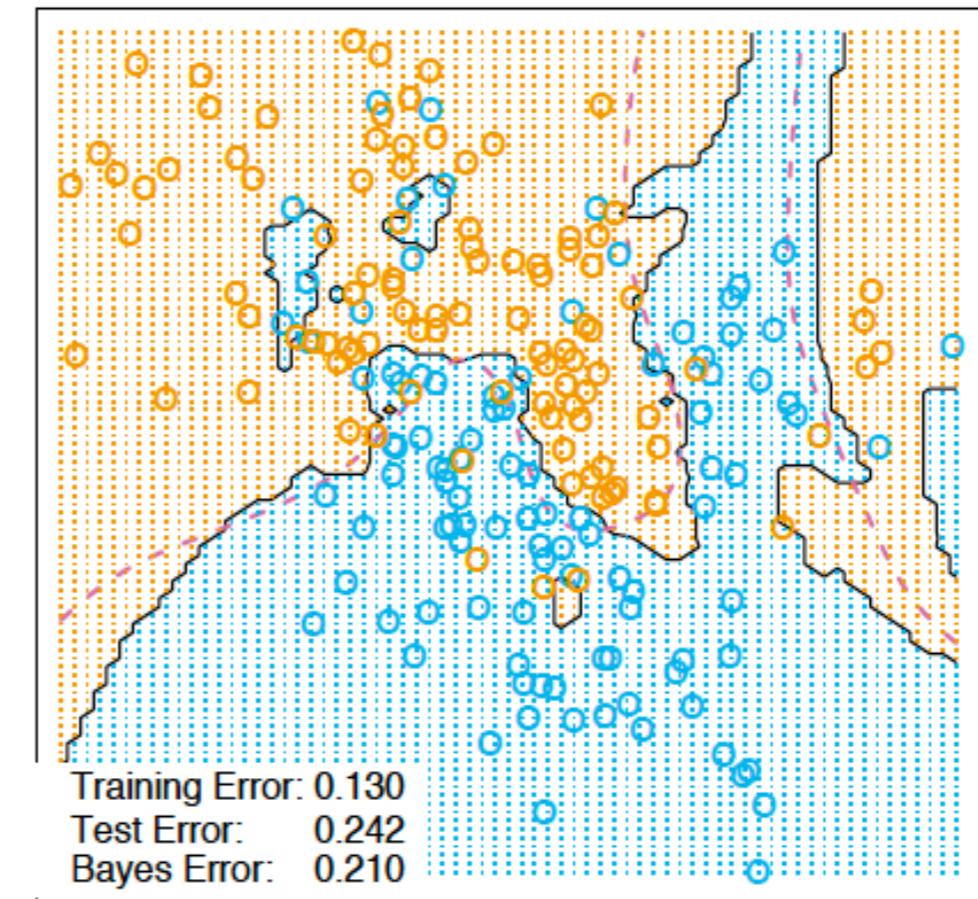
(c) RF (2000 Trees)

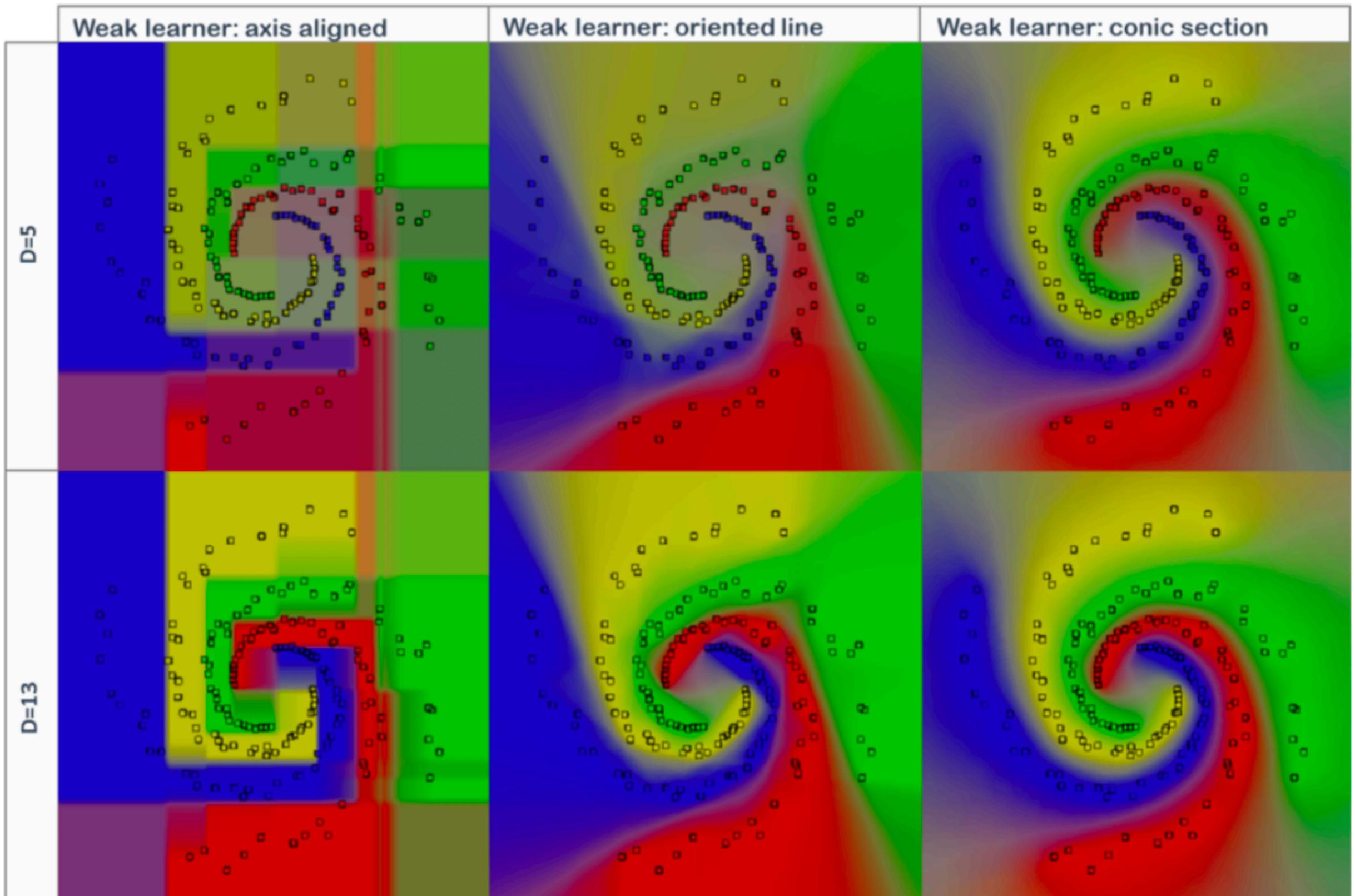
Случайный лес vs kNN

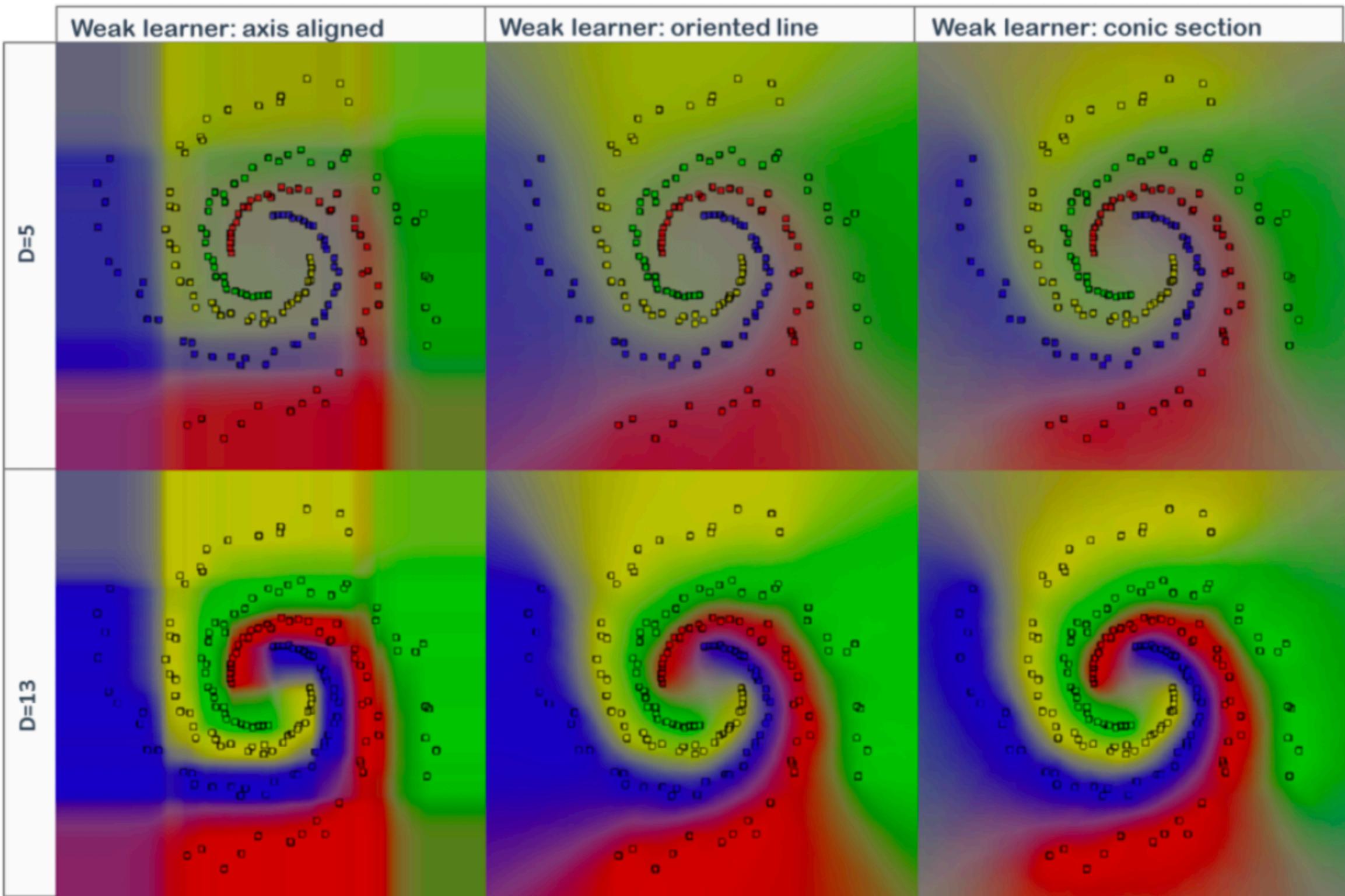
Random Forest Classifier



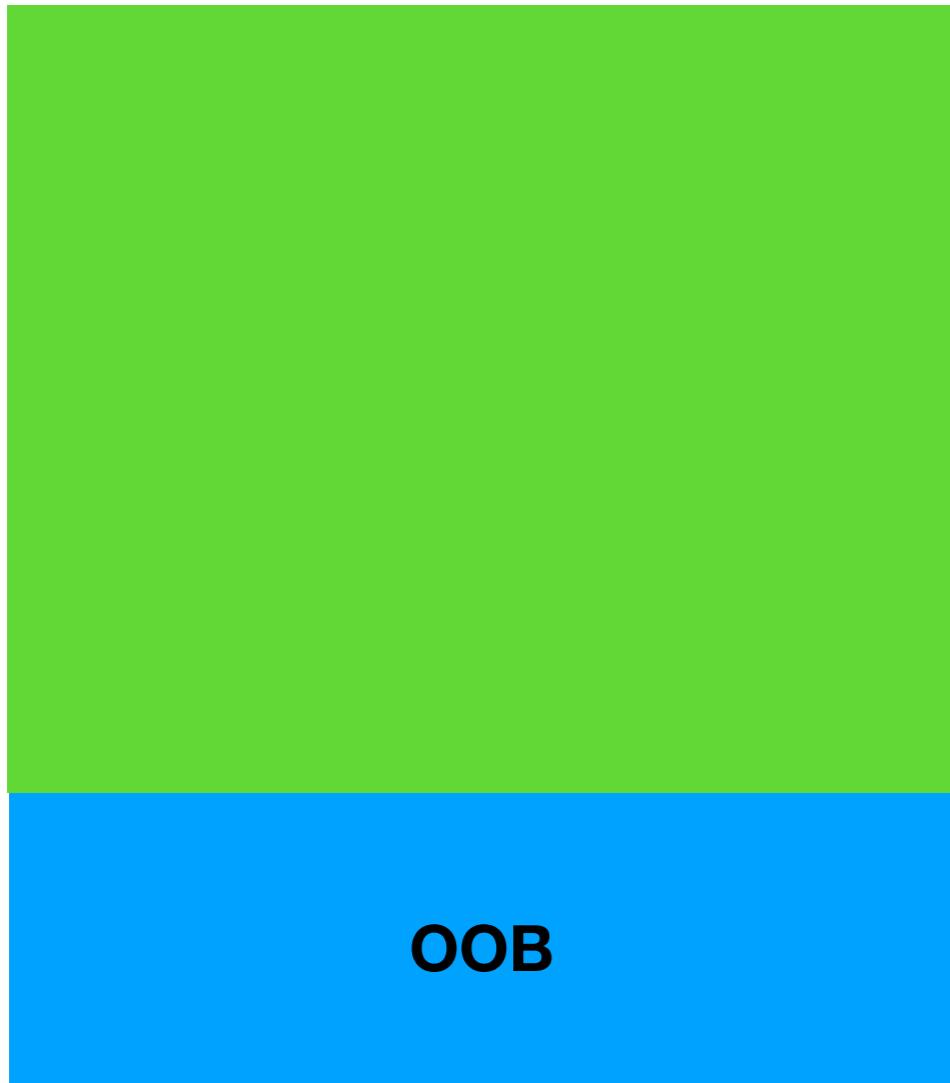
3 Nearest Neighbors





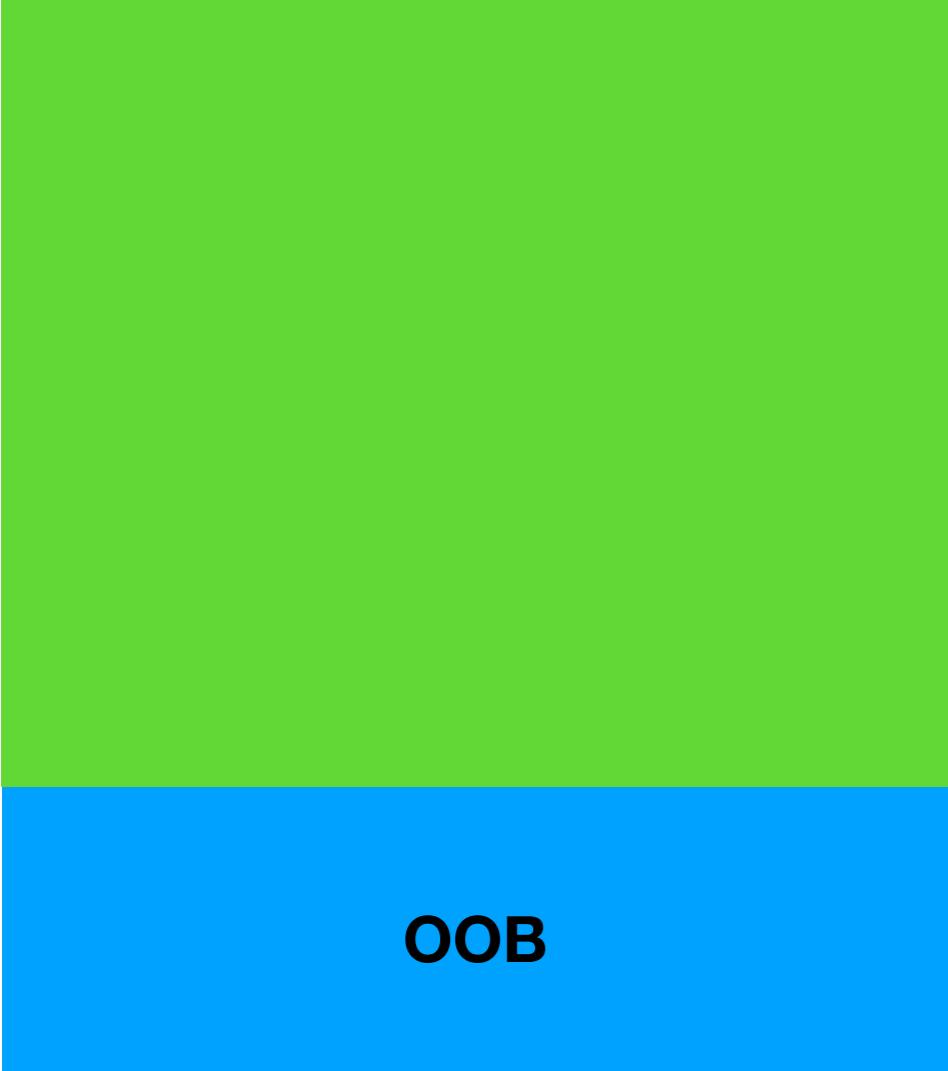


OOB - out of bag samples



**Какая доля объектов не
используется в случайном лесе
для обучения конкретного
дерева?**

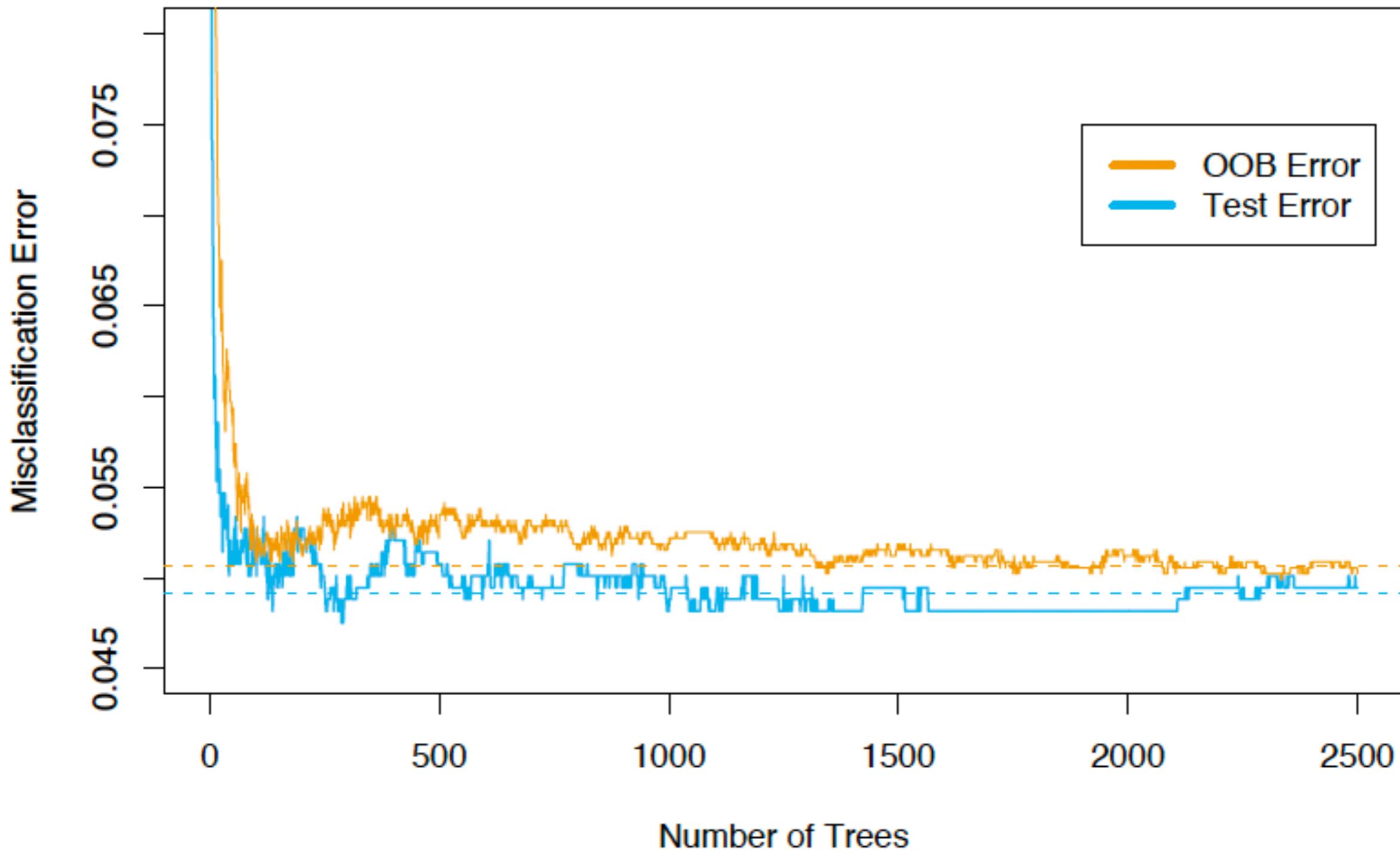
OOB - out of bag samples



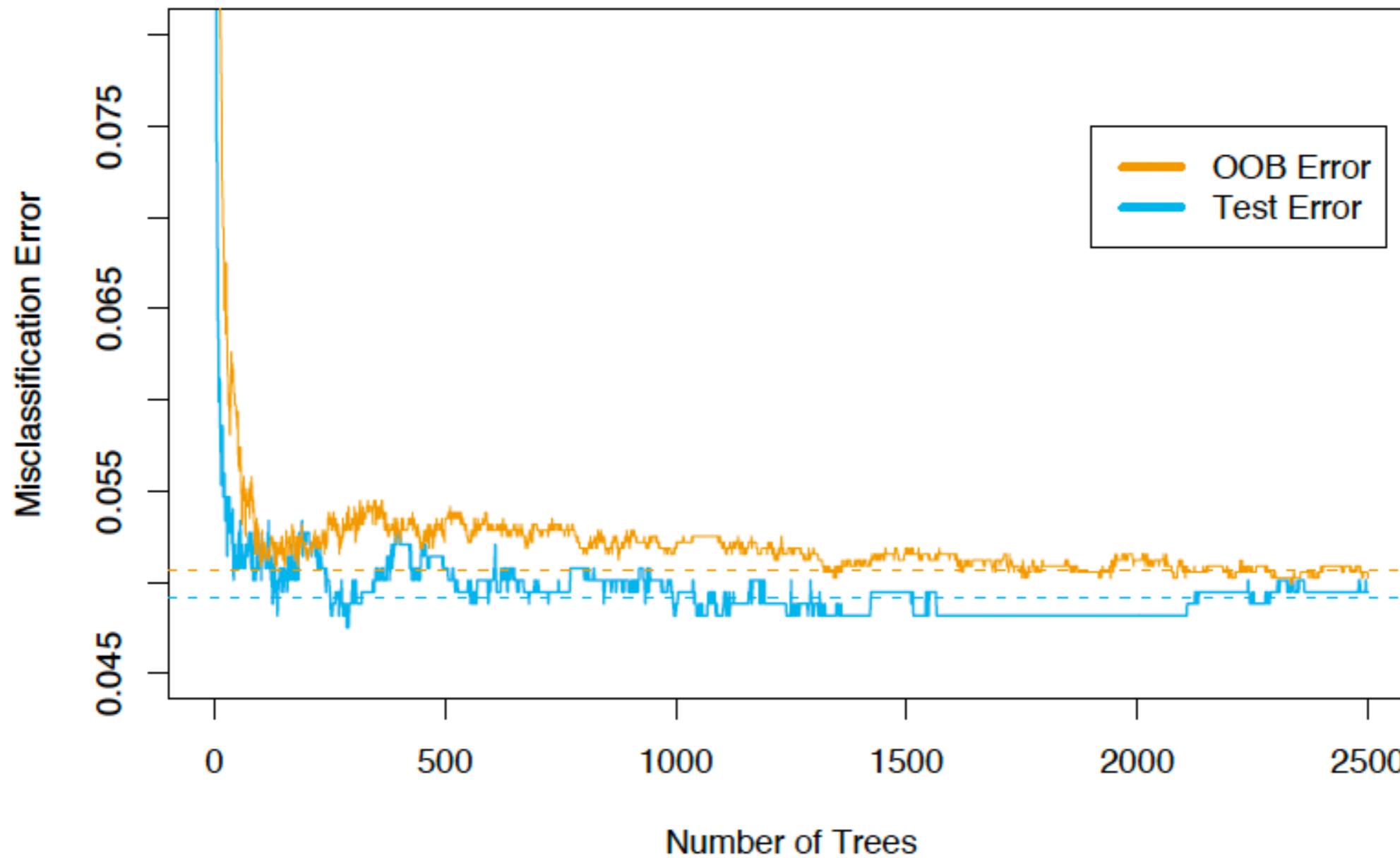
Какая доля объектов используется в случайном лесе для обучения конкретного дерева?

$$\lim_{n \rightarrow \infty} (1 - 1/n)^n = e^{-1}$$

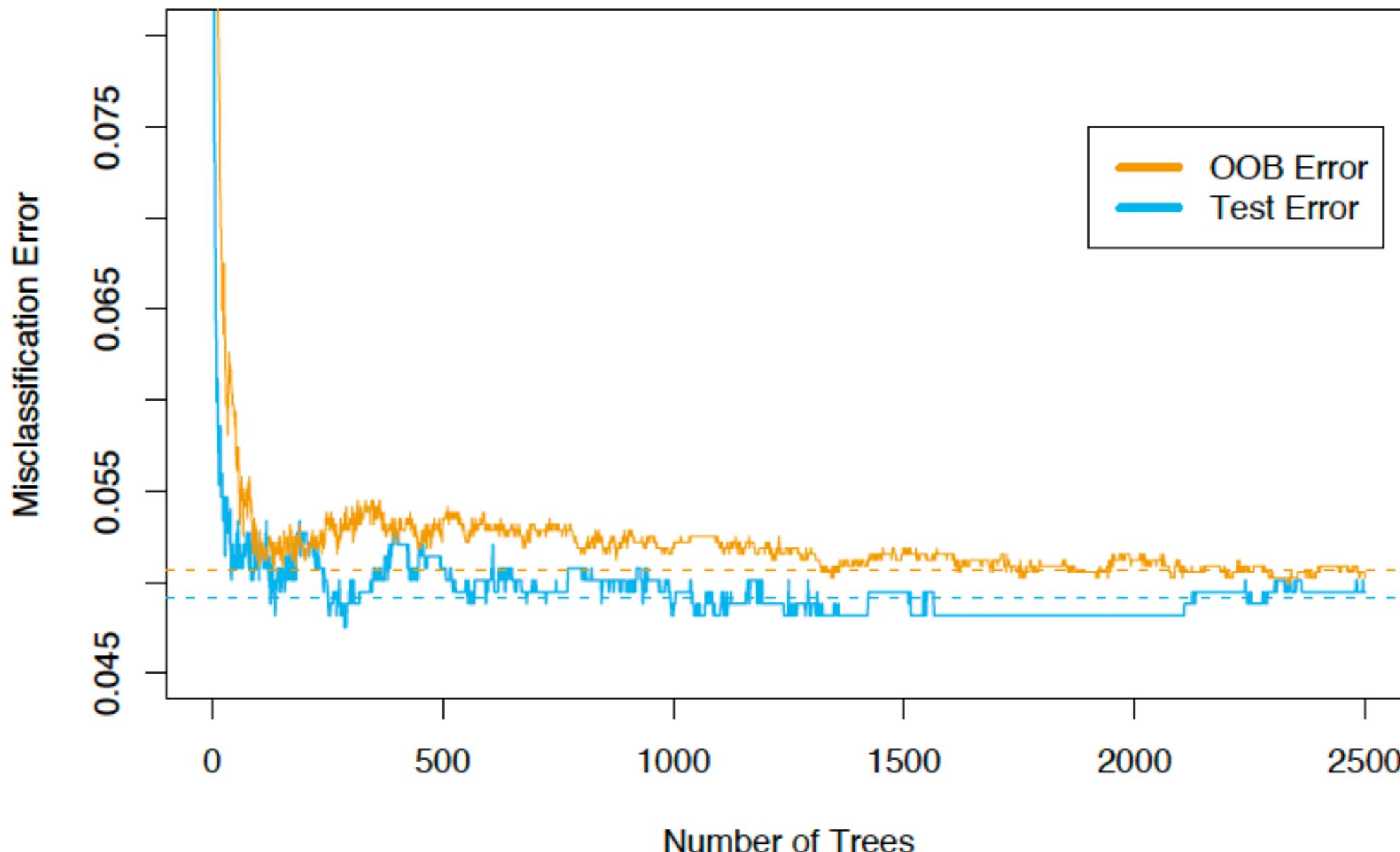
$$1 - 1/e \approx 0.63$$



Переобучается ли случайный лес?



Переобучается ли случайный лес?



С ростом числа деревьев - нет.

Более того, строго говоря, model capacity с ростом числа деревьев уменьшается (по аналогии с kNN)

Переобучается ли случайный лес?

А так - да!

$$\hat{f}_{\text{rf}}(x) = \mathbb{E}_{\Theta} T(x; \Theta) = \lim_{B \rightarrow \infty} \hat{f}(x)_{\text{rf}}^B$$

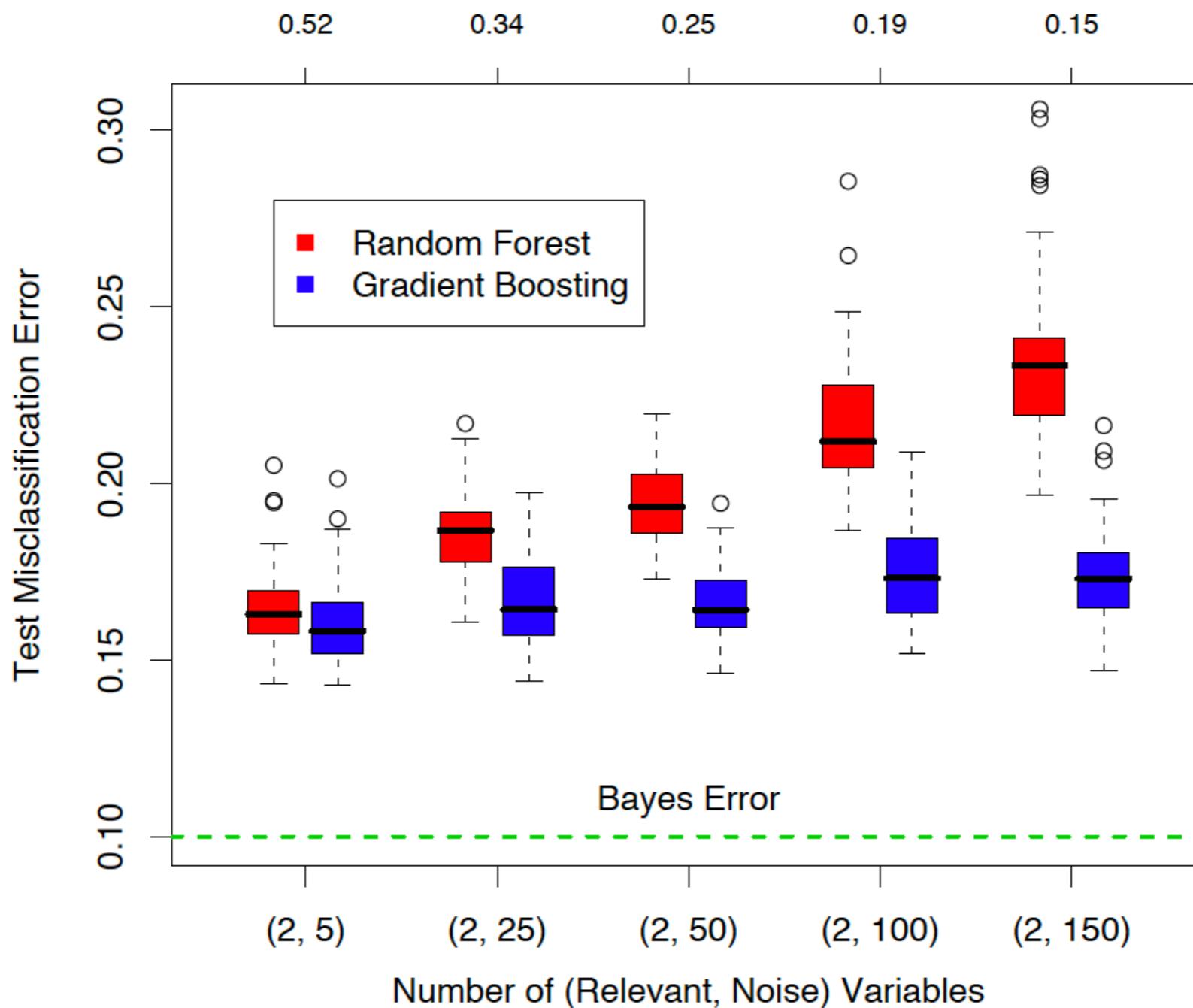
Эта хитрая формула говорит нам, что в пределе случайный лес дает нам качество “идеального” дерева, построенного на наших данных.

И чем больше деревьев мы берем, тем более мы близки.

Но этот идеал как раз может быть переобучен!

- 1) Много бессмысленных признаков
- 2) Утечка данных
- 3) И многое другое

Много бессмысленных переменных



Просто эффект слишком большой глубины

