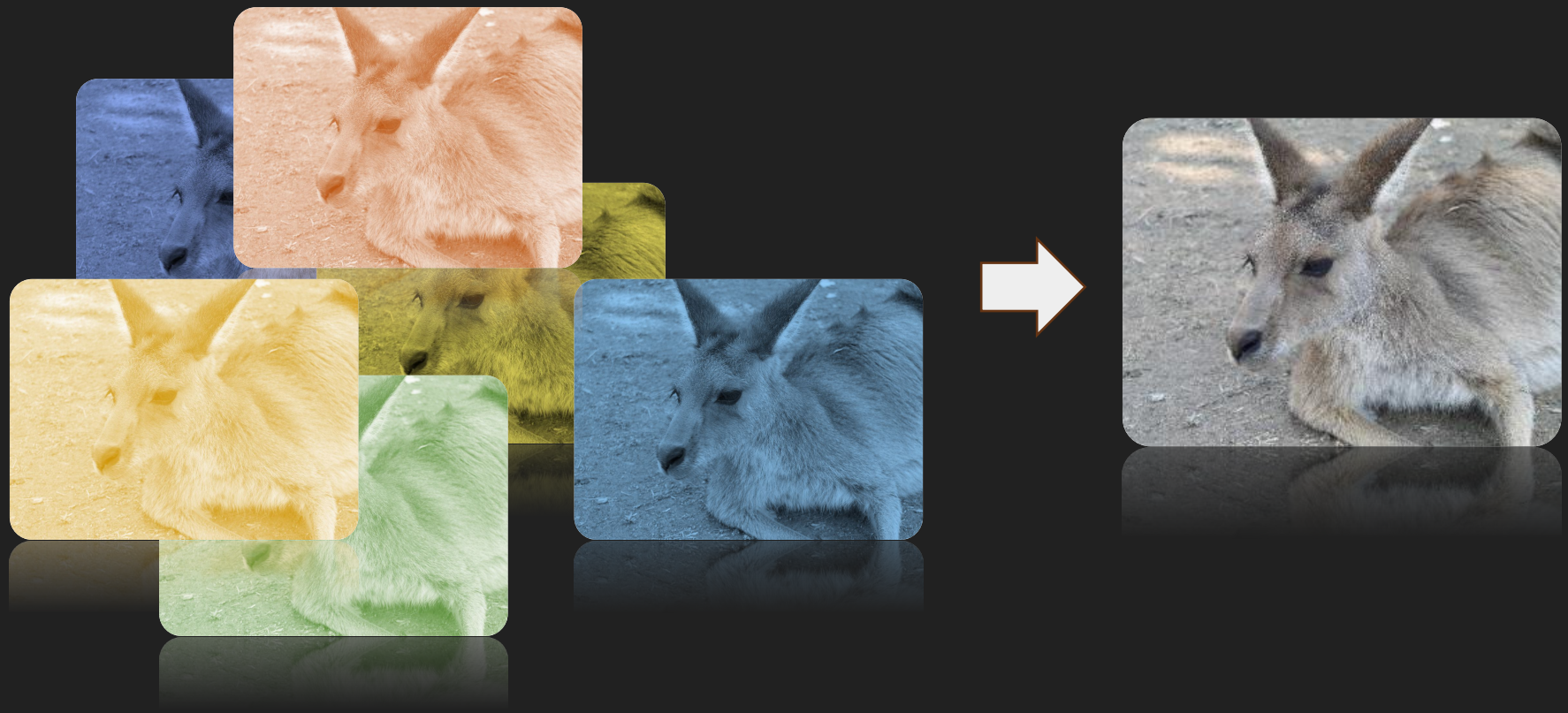
Updates

- Assignment 4 release Wednesday, April 1st DUE 23:59 on April 8th

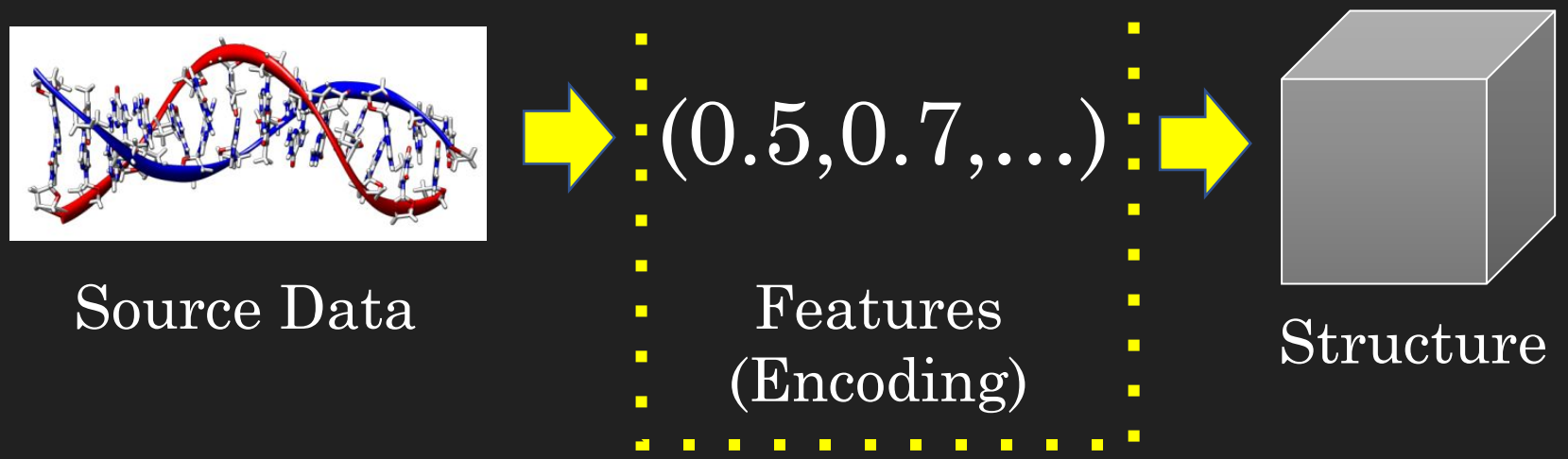
- Literature written report DUE 23:59 on April 8th

- Presentation schedule released: April 6th and 8th during class

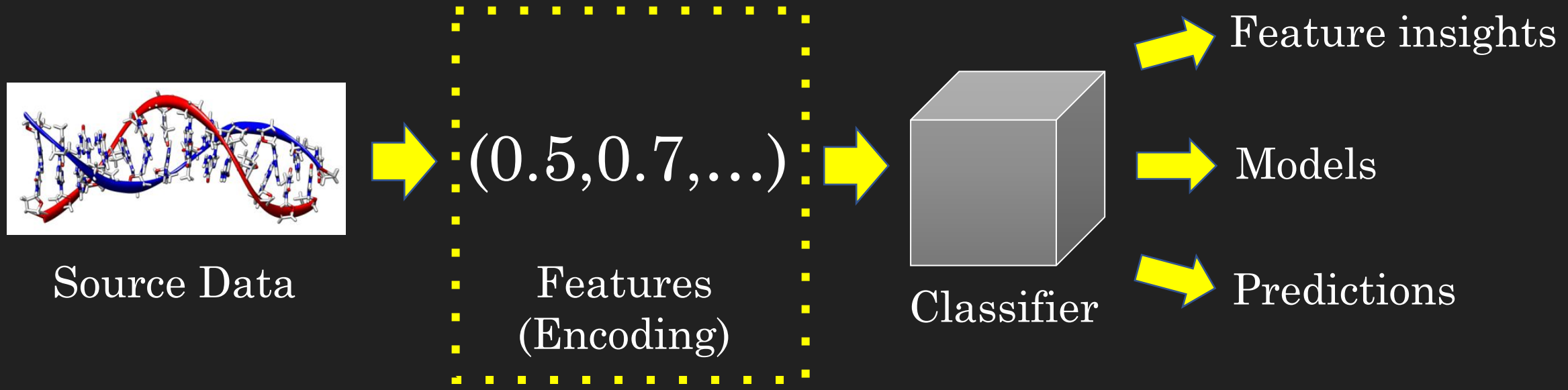


Module 4 Part 2: Features

The big (unsupervised) picture



The big (supervised) picture



Find parameters that define a good decision boundary

Definitions

Feature engineering: The process of turning raw data into useful and efficient representations for ML analysis

Feature selection: Choosing the best subset of existing features

Feature extraction: Representing relevant information in features, which may involve transforming and combining features to generate new representations

Let's represent some DNA!

Reminder: *k*-mers

Decompose a sequence into a set of words of a given length

k-mers: the collection of words of a given length *k*

Nucleotides (k=1): {A,C,G,T}

Dinucleotides (k=2): {AA,AC,…,TT}

Trinucleotides (k=3): {AAA,AAC,…,TTT}

etc…

$N(k)=4^k$

Degenerate Nucleotide Representations

All possible degenerate characters of length 1 to (say) 10

{ A, B, C, …, V } 15^1
{ AA, AB, …, VV } 15^2
…
{ AAAAAAAAAA, AAAAAAAAAC, …, VVVVVVVVVV } 15^{10}

$$15^1 + 15^2 + 15^3 + 15^4 + 15^5 + 15^6 + 15^7 + 15^8 + 15^9 + 15^{10} \approx 5.8 \times 10^{11}$$

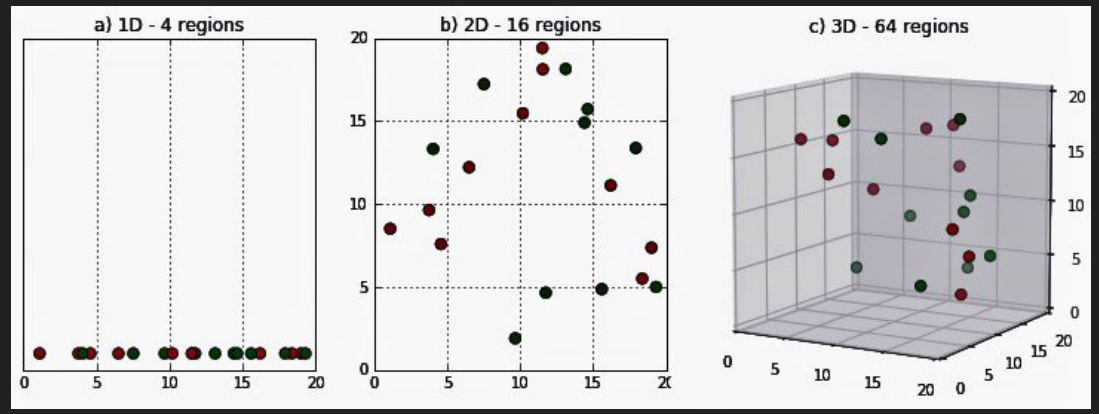
https://www.theguardian.com/us-news/2026/mar/24/amputee-cornhole-player-arrested

Problem?

An excessively high-dimensional set of features / parameters is:

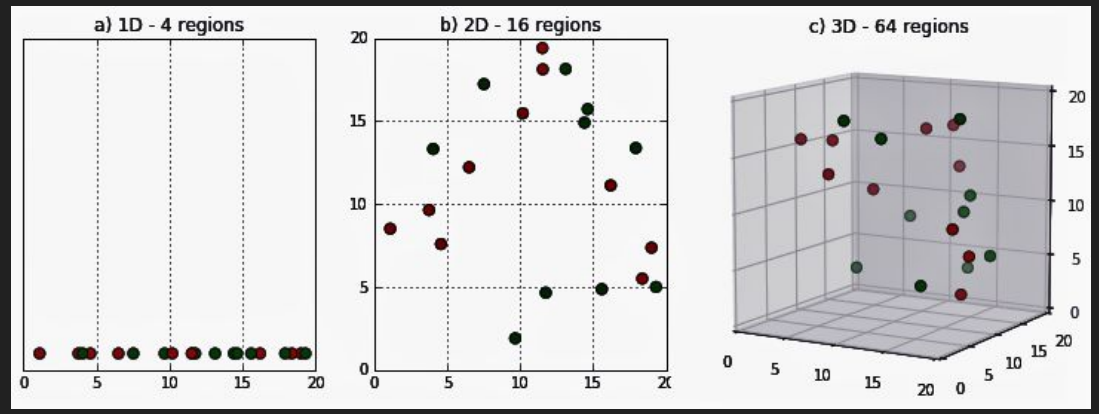
- Computationally intractable
- Fertile ground for overfitting
- Hard to understand!
- Counter-intuitive

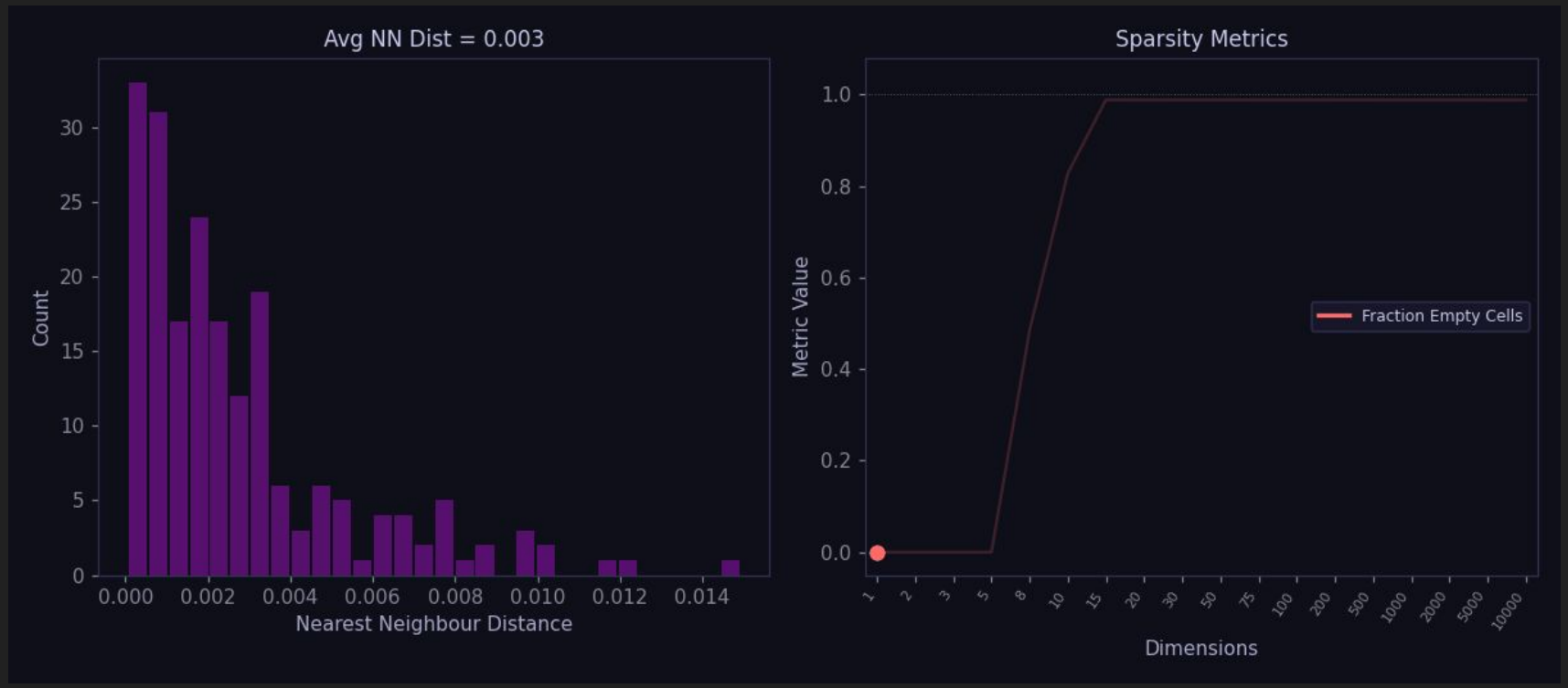
Data becomes increasingly sparse in high dimensions



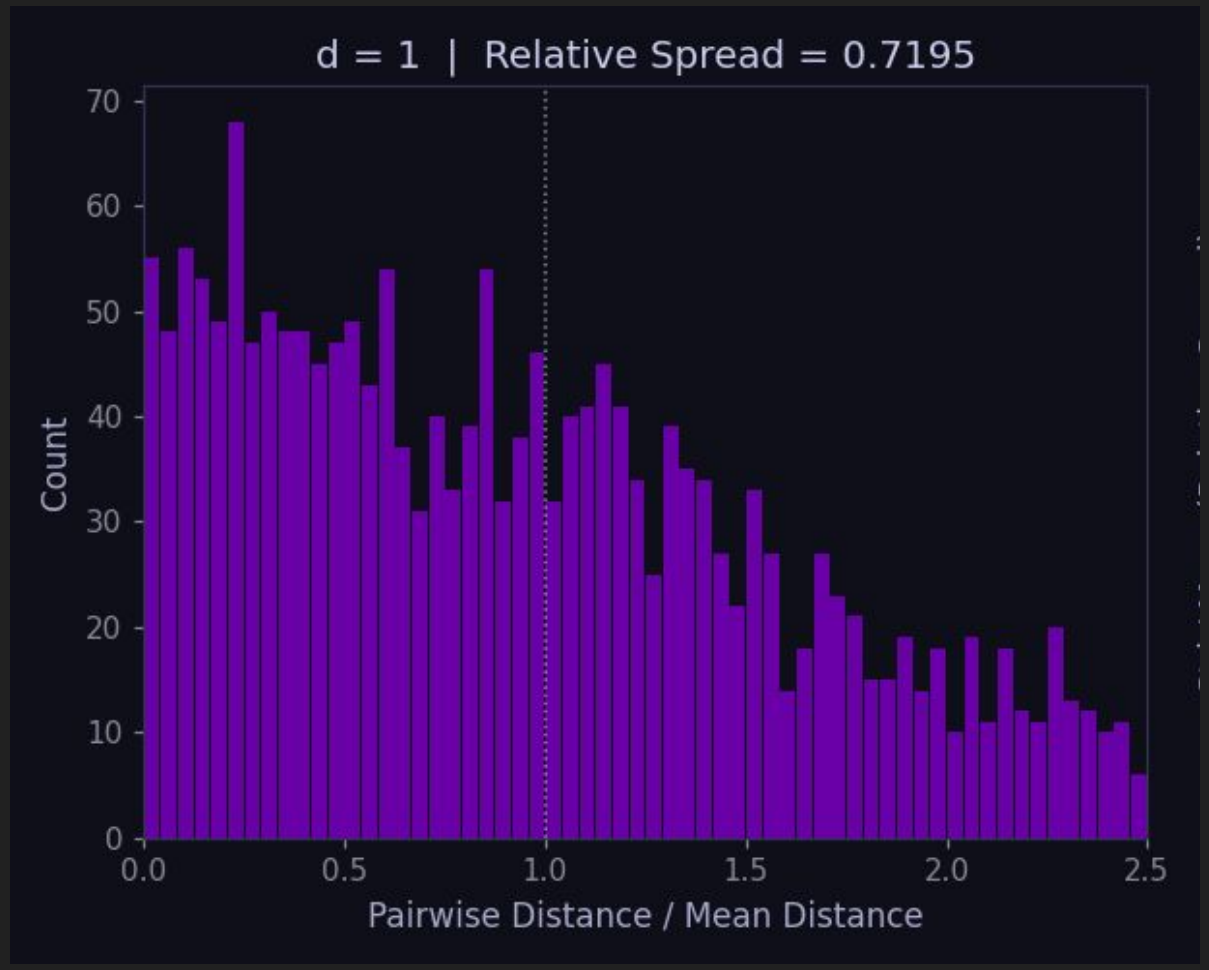
https://medium.com/analytics-vidhya/the-curse-of-dimensionality-and-its-cure-f9891ab72e5c

Data becomes increasingly sparse in high dimensions

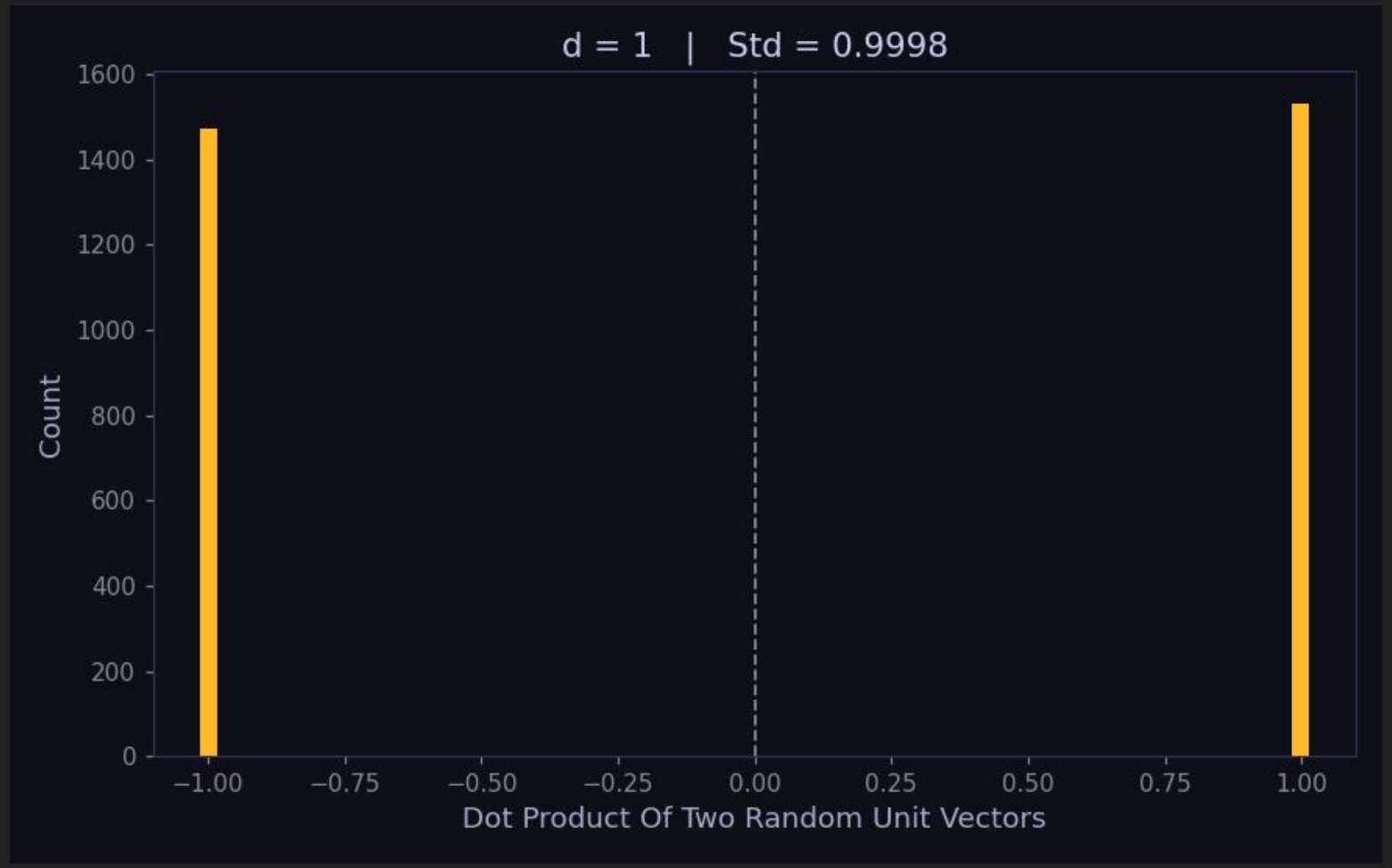


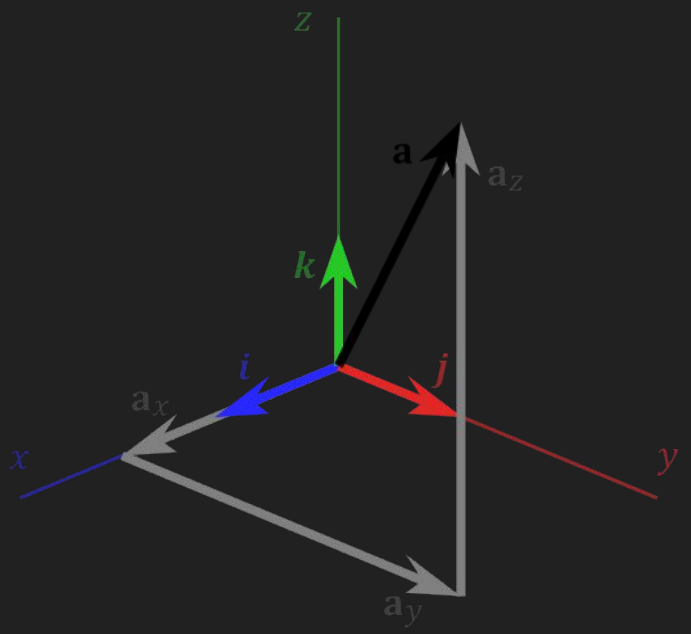


All points look equally far apart from one another

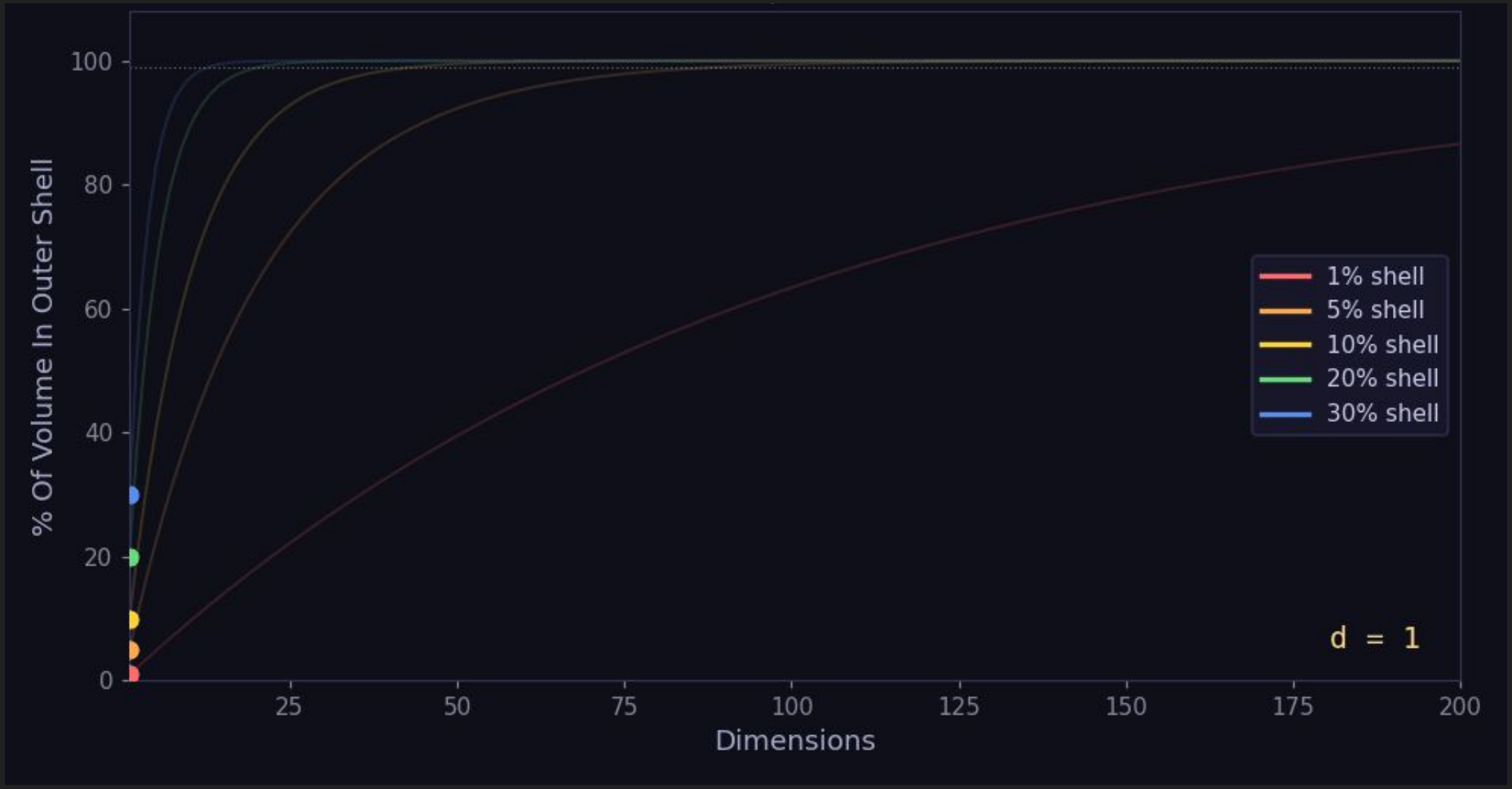


Everything becomes (near) orthogonal

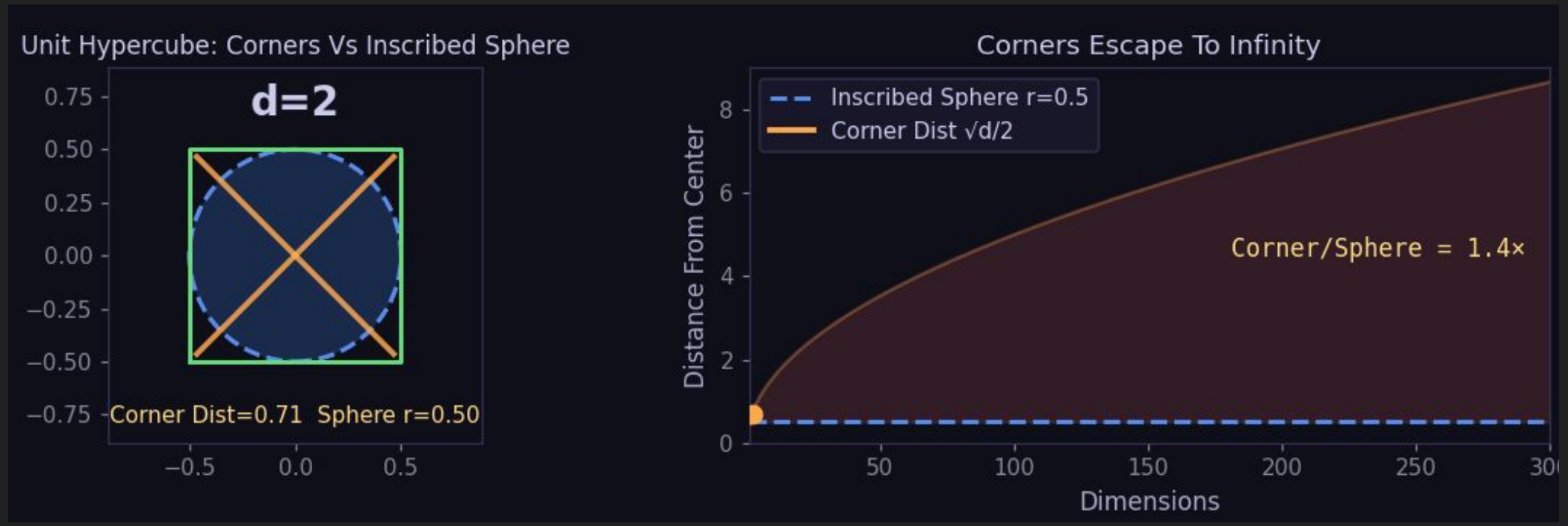




Weird distributions: hypersphere volume concentrates on surface



Weird distributions: Hypercubes get infinitely spiky



So, how do we deal with this?

Dimensionality Reduction: One ticket to model simplification

Define range of representations
(e.g.
compositional vectors up to size k,
Markov models up to size m,
structural features)

Identify individual features that are most useful for classification

Extract essential information from sets of features

Feature SELECTION

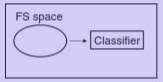
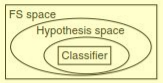
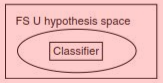
Feature EXTRACTION

Classification technique

Feature Selection

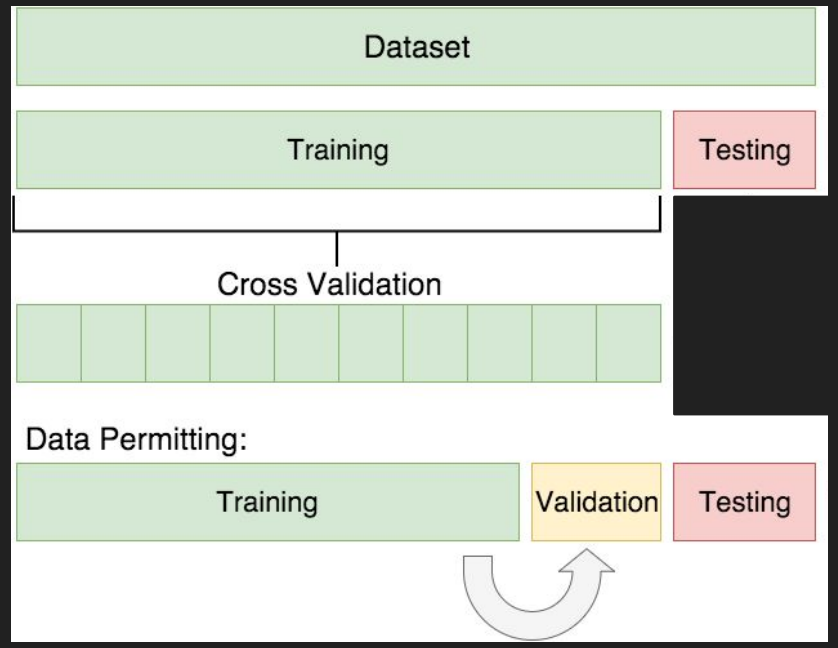
Differ in the role of the classifier

- Filter: select based on a simple criterion
- Wrapper: select based on their performance in the classifier
- Embedded: select during optimization process

Model search	Advantages	Disadvantages	Examples
Filter	Univariate		
	Fast Scalable Independent of the classifier	Ignores feature dependencies Ignores interaction with the classifier	χ^2 Euclidean distance *i*-test Information gain, Gain ratio (Ben-Bassat, 1982)
	Multivariate		
	Models feature dependencies Independent of the classifier Better computational complexity than wrapper methods	Slower than univariate techniques Less scalable than univariate techniques Ignores interaction with the classifier	Correlation-based feature selection (CFS) (Hall, 1999) Markov blanket filter (MBF) (Koller and Sahami, 1996) Fast correlation-based feature selection (FCBF) (Yu and Liu, 2004)
Wrapper	Deterministic		
	Simple Interacts with the classifier Models feature dependencies Less computationally intensive than randomized methods	Risk of over fitting More prone than randomized algorithms to getting stuck in a local optimum (greedy search) Classifier dependent selection	Sequential forward selection (SFS) (Kittler, 1978) Sequential backward elimination (SBE) (Kittler, 1978) Plus *q* take-away *r* (Ferri *et al.*, 1994) Beam search (Siedelecky and Sklansky, 1988)
	Randomized		
	Less prone to local optima Interacts with the classifier Models feature dependencies	Computationally intensive Classifier dependent selection Higher risk of overfitting than deterministic algorithms	Simulated annealing Randomized hill climbing (Skalak, 1994) Genetic algorithms (Holland, 1975) Estimation of distribution algorithms (Inza *et al.*, 2000)
Embedded	Interacts with the classifier Better computational complexity than wrapper methods Models feature dependencies	Classifier dependent selection	Decision trees Weighted naive Bayes (Duda *et al.*, 2001) Feature selection using the weight vector of SVM (Guyon *et al.*, 2002; Weston *et al.*, 2003)
			

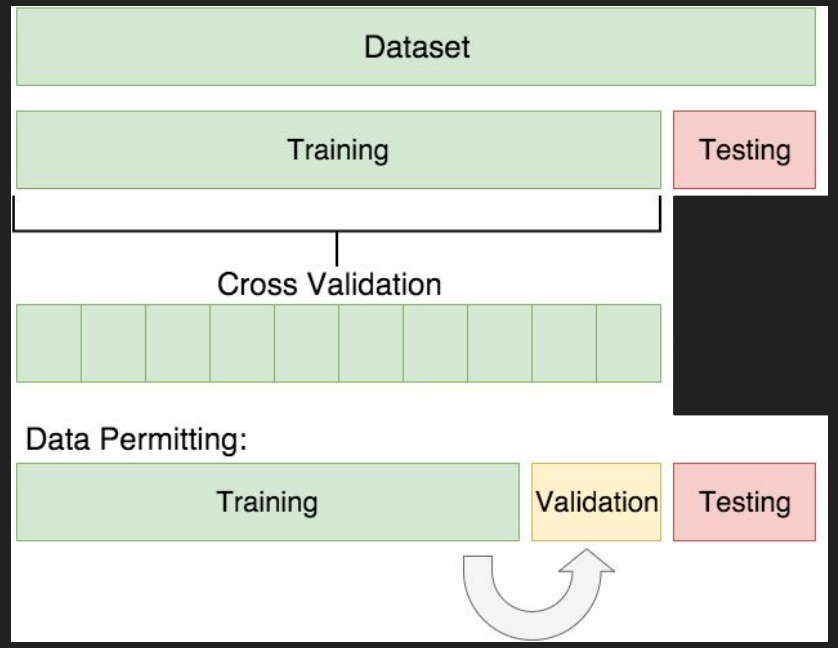
Very easy to leak from train to test in feature selection

Patient ID	Age	Sex	BMI	Systolic BP (mmHg)	Diastolic BP (mmHg)	Fasting Glucose (mmol/L)	HbA1c (%)	Cholesterol (mmol/L)	Smoker	Family History Diabetes	Diagnosis (Classification)	10-yr CVD Risk (Regression)
P001	54	M	28.3	138	88	6.2	6.8	5.1	Yes	Yes	Diabetic	18.4%
P002	41	F	22.1	118	75	4.9	5.2	4.7	No	No	Healthy	4.1%
P003	67	M	31.7	155	95	7.8	7.4	6.3	Yes	Yes	Diabetic	31.2%
P004	35	F	25.6	122	80	5.3	5.5	4.2	No	Yes	Healthy	5.8%
P005	58	M	29.9	145	91	6.9	6.5	5.8	No	No	Diabetic	14.7%
P006	72	F	27.4	160	97	8.4	8.1	6.9	Yes	Yes	Diabetic	38.5%
P007	29	M	23.8	115	73	4.7	5	4	No	No	Healthy	2.3%
P008	63	F	33.2	148	93	7.1	7	5.5	No	Yes	Diabetic	25.6%



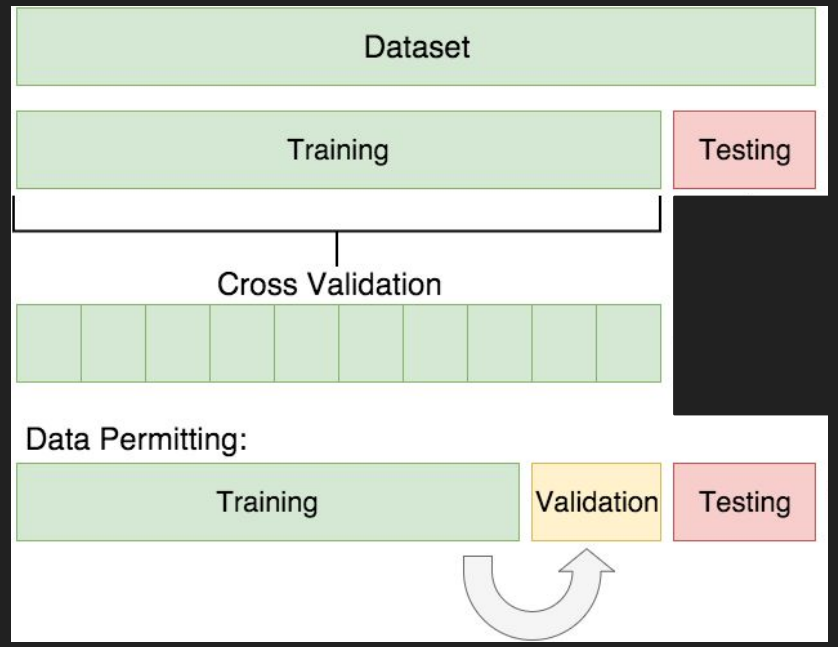
Very easy to leak from train to test in feature selection

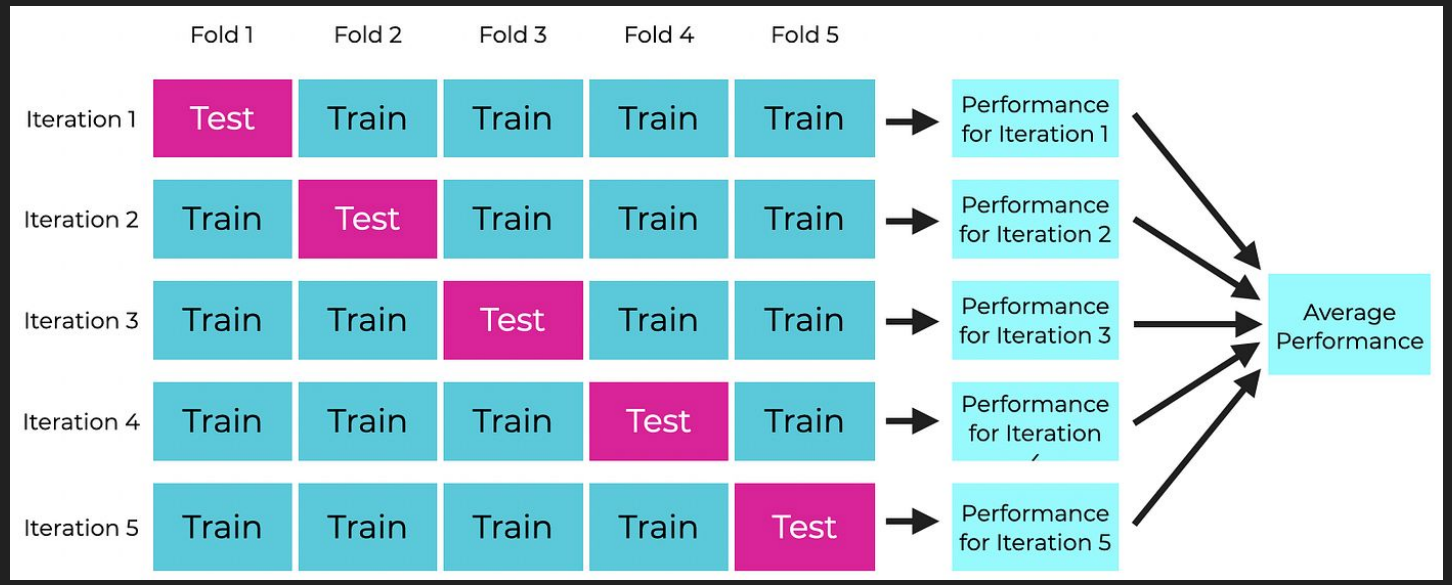
Patient ID	Age	Sex	BMI	Systolic BP (mmHg)	Diastolic BP (mmHg)	Fasting Glucose (mmol/L)	HbA1c (%)	Cholesterol (mmol/L)	Smoker	Family History Diabetes	Diagnosis (Classification)	10-yr CVD Risk (Regression)
P001	54	M	28.3	138	88	6.2	6.8	5.1	Yes	Yes	Diabetic	18.4%
P003	67	M	31.7	155	95	7.8	7.4	6.3	Yes	Yes	Diabetic	31.2%
P004	35	F	25.6	122	80	5.3	5.5	4.2	No	Yes	Healthy	5.8%
P006	72	F	27.4	160	97	8.4	8.1	6.9	Yes	Yes	Diabetic	38.5%
P007	29	M	23.8	115	73	4.7	5	4	No	No	Healthy	2.3%
P008	63	F	33.2	148	93	7.1	7	5.5	No	Yes	Diabetic	25.6%



Very easy to leak from train to test in feature selection

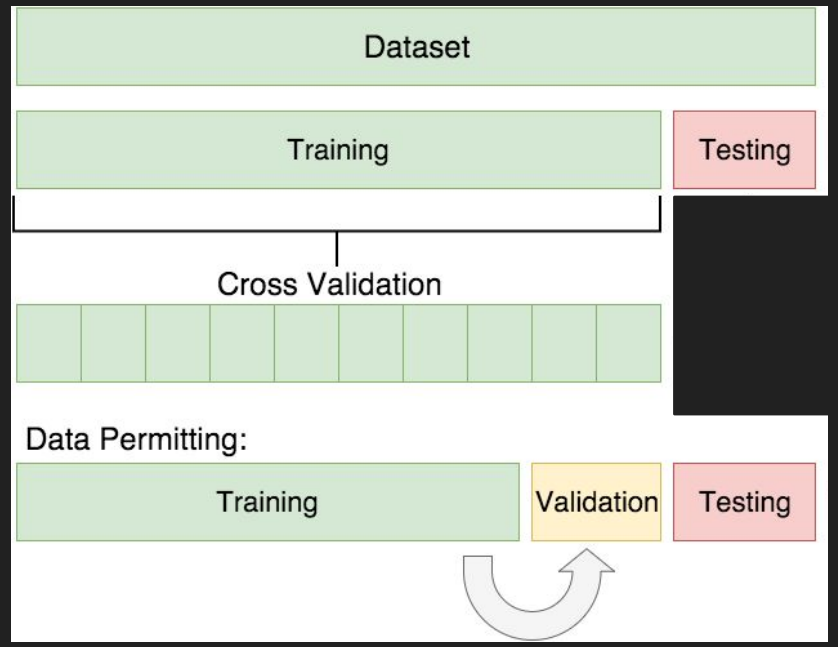
Patient ID	Age	Sex	BMI	Systolic BP (mmHg)	Diastolic BP (mmHg)	Fasting Glucose (mmol/L)	HbA1c (%)	Cholesterol (mmol/L)	Smoker	Family History Diabetes	Diagnosis (Classification)	10-yr CVD Risk (Regression)
P001	54	M	28.3	138	88	6.2	6.8	5.1	Yes	Yes	Diabetic	18.4%
P003	67	M	31.7	155	95	7.8	7.4	6.3	Yes	Yes	Diabetic	31.2%
P004	35	F	25.6	122	80	5.3	5.5	4.2	No	Yes	Healthy	5.8%
P006	72	F	27.4	160	97	8.4	8.1	6.9	Yes	Yes	Diabetic	38.5%
P007	29	M	23.8	115	73	4.7	5	4	No	No	Healthy	2.3%
P008	63	F	33.2	148	93	7.1	7	5.5	No	Yes	Diabetic	25.6%

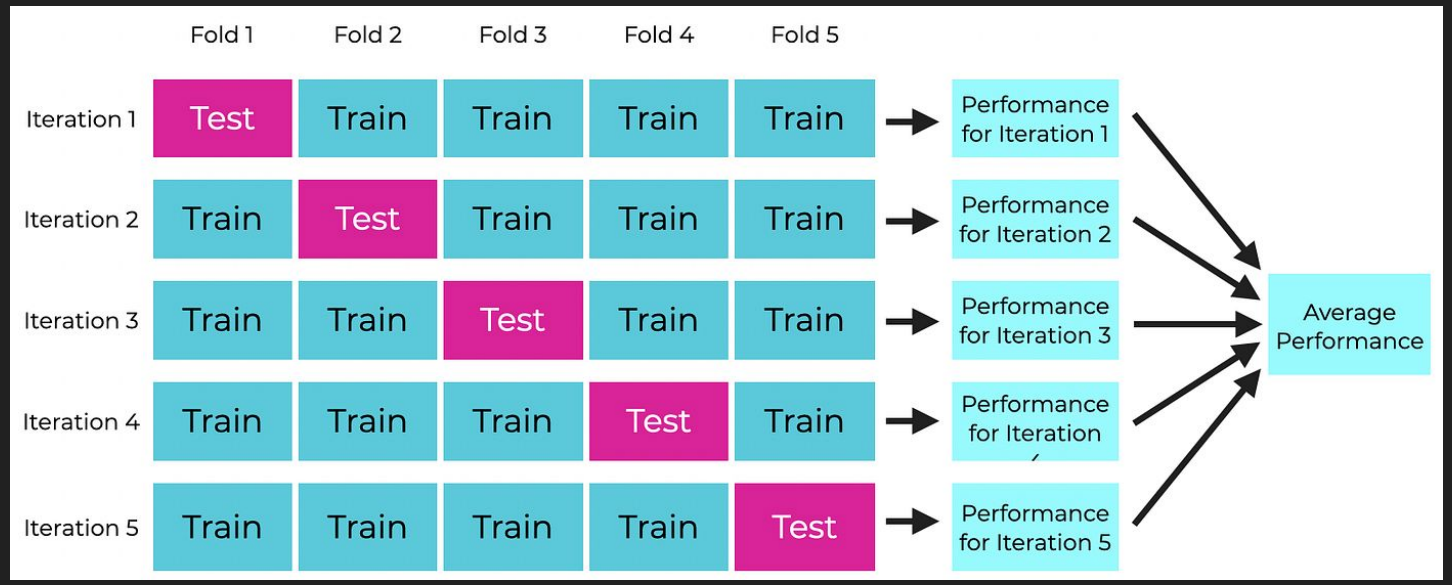




Very easy to leak from train to test in feature selection

Patient ID	Age	Sex	BMI	Systolic BP (mmHg)	Diastolic BP (mmHg)	Fasting Glucose (mmol/L)	HbA1c (%)	Cholesterol (mmol/L)	Smoker	Family History Diabetes	Diagnosis (Classification)	10-yr CVD Risk (Regression)
P001	54	M	28.3	138	88	6.2	6.8	5.1	Yes	Yes	Diabetic	18.4%
P003	67	M	31.7	155	95	7.8	7.4	6.3	Yes	Yes	Diabetic	31.2%
P004	35	F	25.6	122	80	5.3	5.5	4.2	No	Yes	Healthy	5.8%
P006	72	F	27.4	160	97	8.4	8.1	6.9	Yes	Yes	Diabetic	38.5%
P007	29	M	23.8	115	73	4.7	5	4	No	No	Healthy	2.3%
P008	63	F	33.2	148	93	7.1	7	5.5	No	Yes	Diabetic	25.6%

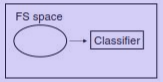
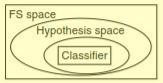
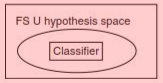




Feature Selection

Differ in the role of the classifier

- Filter: select based on a simple criterion
- Wrapper: select based on their performance in the classifier
- Embedded: select during optimization process

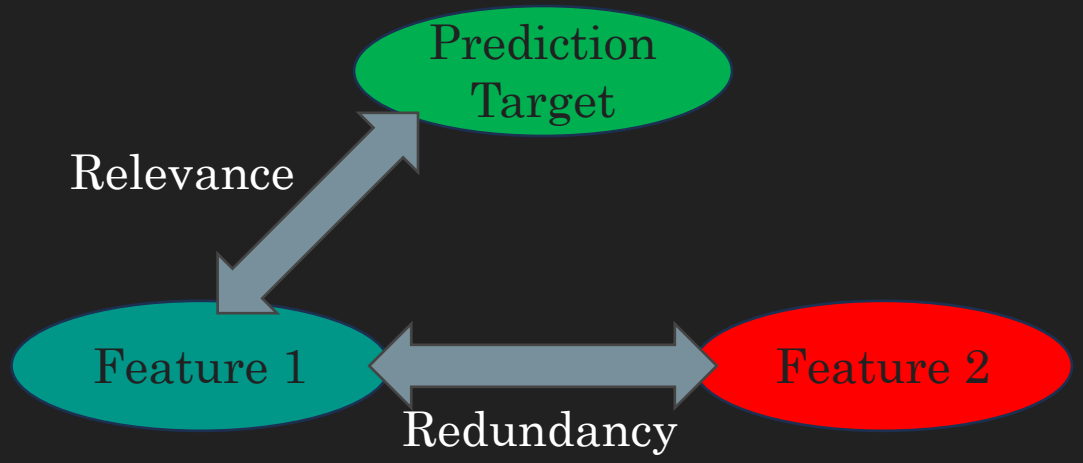
Model search	Advantages	Disadvantages	Examples
Filter	Univariate		
	Fast Scalable Independent of the classifier	Ignores feature dependencies Ignores interaction with the classifier	χ^2 Euclidean distance *i*-test Information gain, Gain ratio (Ben-Bassat, 1982)
	Multivariate		
	Models feature dependencies Independent of the classifier Better computational complexity than wrapper methods	Slower than univariate techniques Less scalable than univariate techniques Ignores interaction with the classifier	Correlation-based feature selection (CFS) (Hall, 1999) Markov blanket filter (MBF) (Koller and Sahami, 1996) Fast correlation-based feature selection (FCBF) (Yu and Liu, 2004)
Wrapper	Deterministic		
	Simple Interacts with the classifier Models feature dependencies Less computationally intensive than randomized methods	Risk of over fitting More prone than randomized algorithms to getting stuck in a local optimum (greedy search) Classifier dependent selection	Sequential forward selection (SFS) (Kittler, 1978) Sequential backward elimination (SBE) (Kittler, 1978) Plus *q* take-away *r* (Ferri *et al.*, 1994) Beam search (Siedelecky and Sklansky, 1988)
	Randomized		
	Less prone to local optima Interacts with the classifier Models feature dependencies	Computationally intensive Classifier dependent selection Higher risk of overfitting than deterministic algorithms	Simulated annealing Randomized hill climbing (Skalak, 1994) Genetic algorithms (Holland, 1975) Estimation of distribution algorithms (Inza *et al.*, 2000)
Embedded	Interacts with the classifier Better computational complexity than wrapper methods Models feature dependencies	Classifier dependent selection	Decision trees Weighted naive Bayes (Duda *et al.*, 2001) Feature selection using the weight vector of SVM (Guyon *et al.*, 2002; Weston *et al.*, 2003)
			

Filter Methods

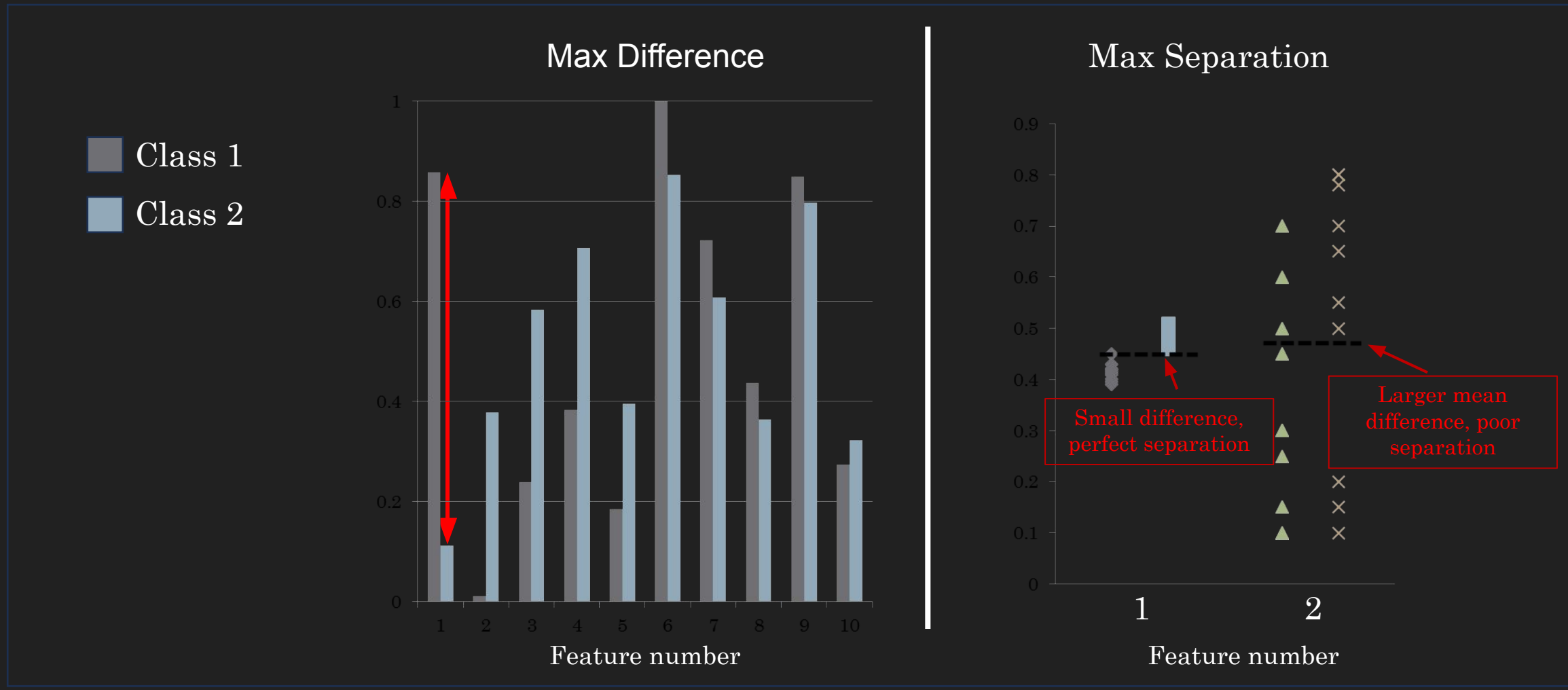
Consider the individual impact of features *before* using the classifier (typically using a simple screening criterion)

Feature RELEVANCE

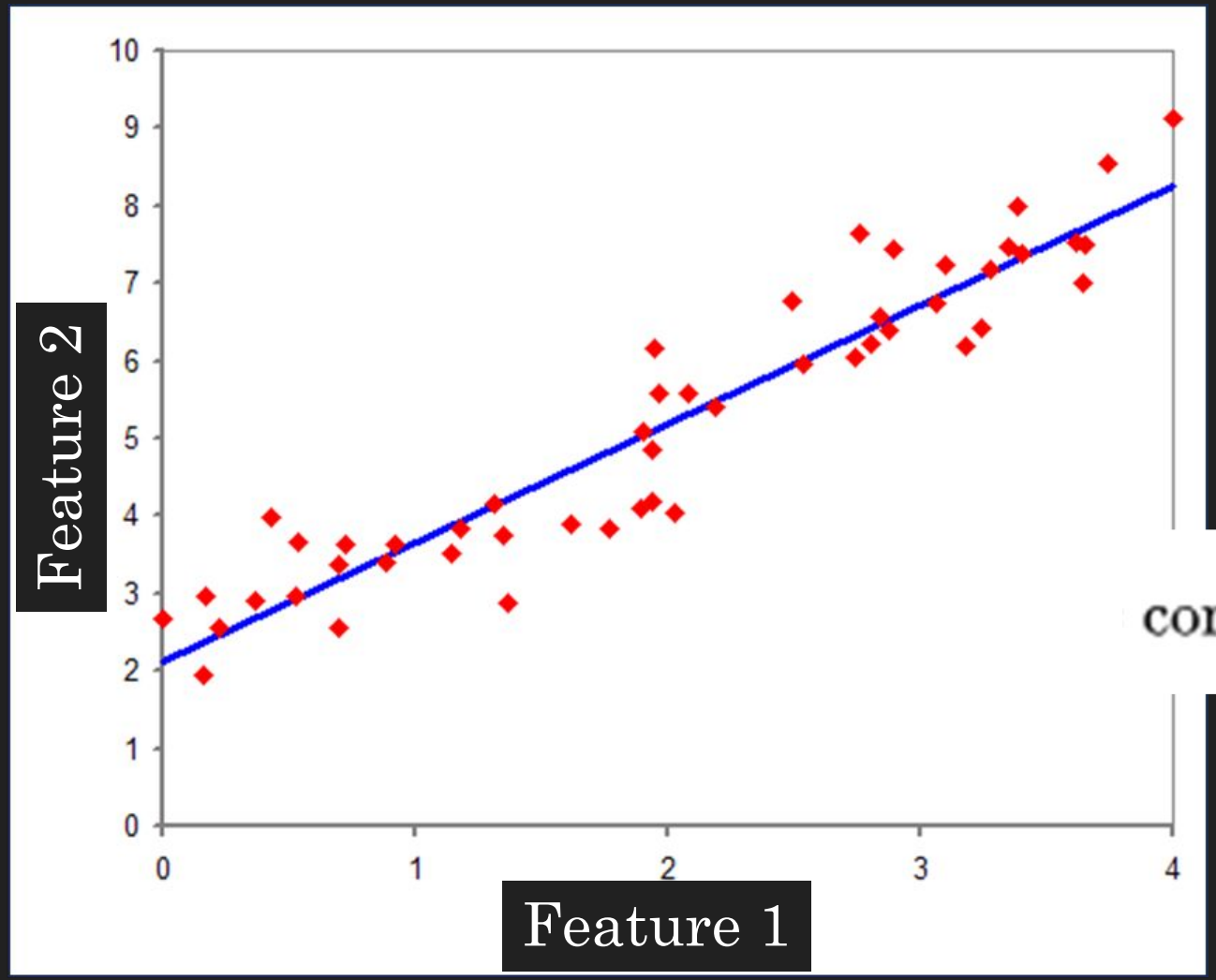
Feature REDUNDANCY



Relevance


Class 1
Class 2
Max Difference
1
0.8
0.6
0.4
0.2
0
1 2 3 4 5 6 7 8 9 10
Feature number
Max Separation
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1 2
Small difference, perfect separation
Larger mean difference, poor separation
Feature number

Correlation:
Redundancy of continuous-valued features



$$\operatorname{corr}(X,Y) = \frac{\mathrm{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

Difference between X value and mean X

Difference between Y value and mean Y

Standard deviations

(Equation cribbed from Wikipedia)

Mutual Information: Redundancy of categorical-valued features

For two categorical features X and Y:

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

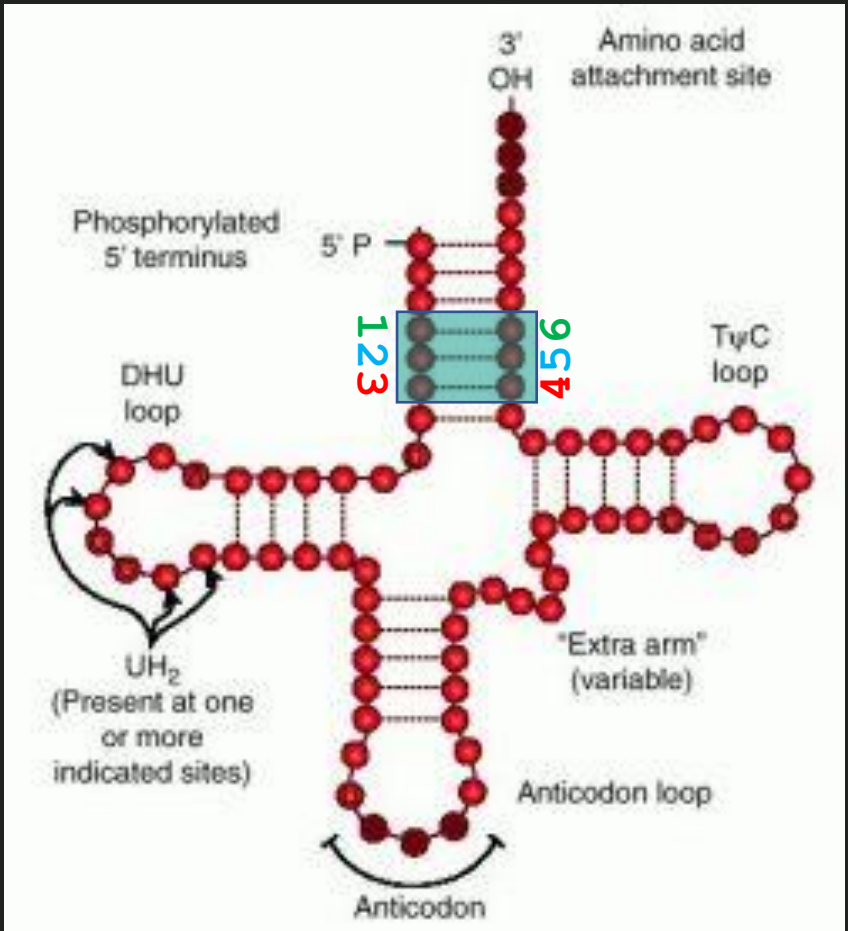
Probability that two classes are seen together in this dataset

Independent probabilities of each class

How well does y predict the value of x (or vice versa?)

Also applicable to continuous-valued features (integrals)

Mutual information in tRNA Sequences



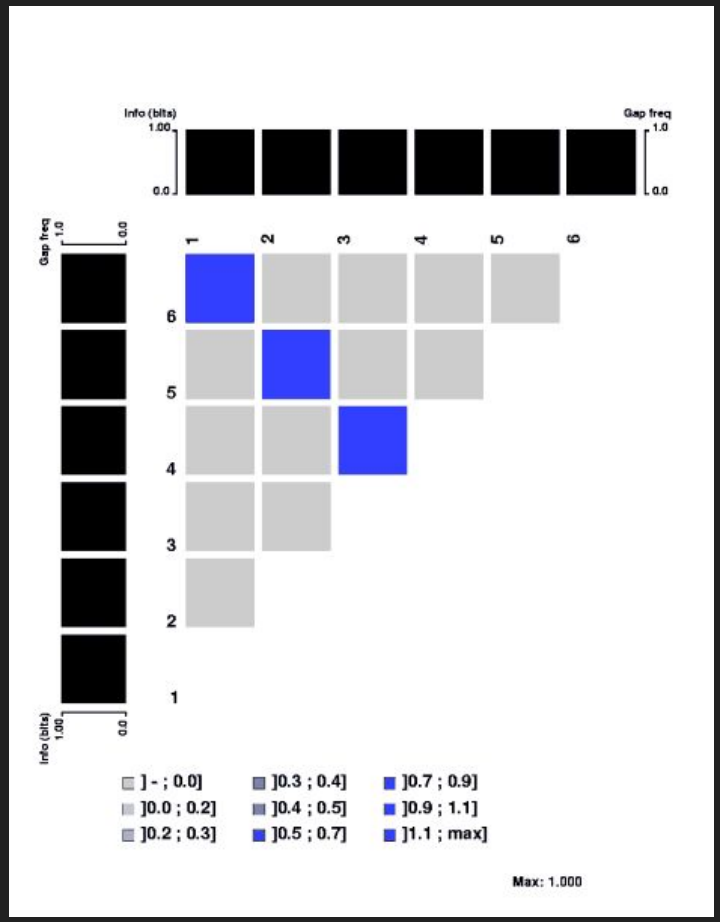
123 456

(1) UCG…CGA

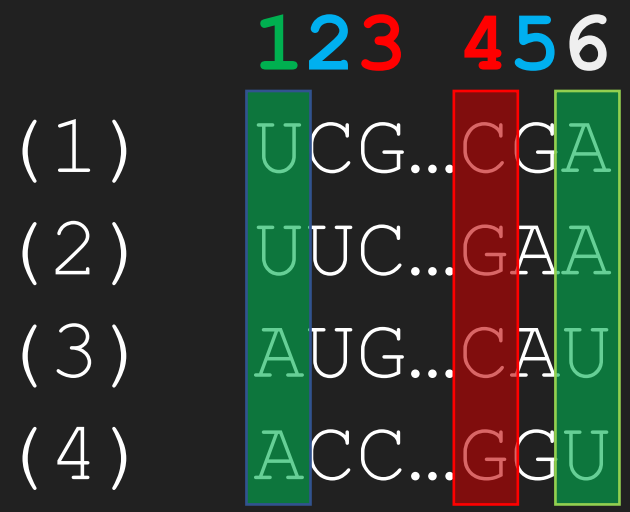
(2) UUC…GAA

(3) AUG…CAU

(4) ACC…GGU



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



col 1 vs col 4

col 1 vs col 6

I = $4[0.25 \times \log_2(0.25 / 0.25)]$

= 0

(complete independence)

I = $2[0.5 \times \log_2(0.5 / 0.25)]$

= 1

(complete redundancy)

Minimum Redundancy – Maximum Relevance (MRMR)

Minimum redundancy: select features that are largely independent, as assessed by

- Low mutual information
- Minimal correlation
- Maximal Euclidean distance

Maximum relevance: select features that are good classifiers!

MRMR aims to maximize either

(relevance – redundancy)

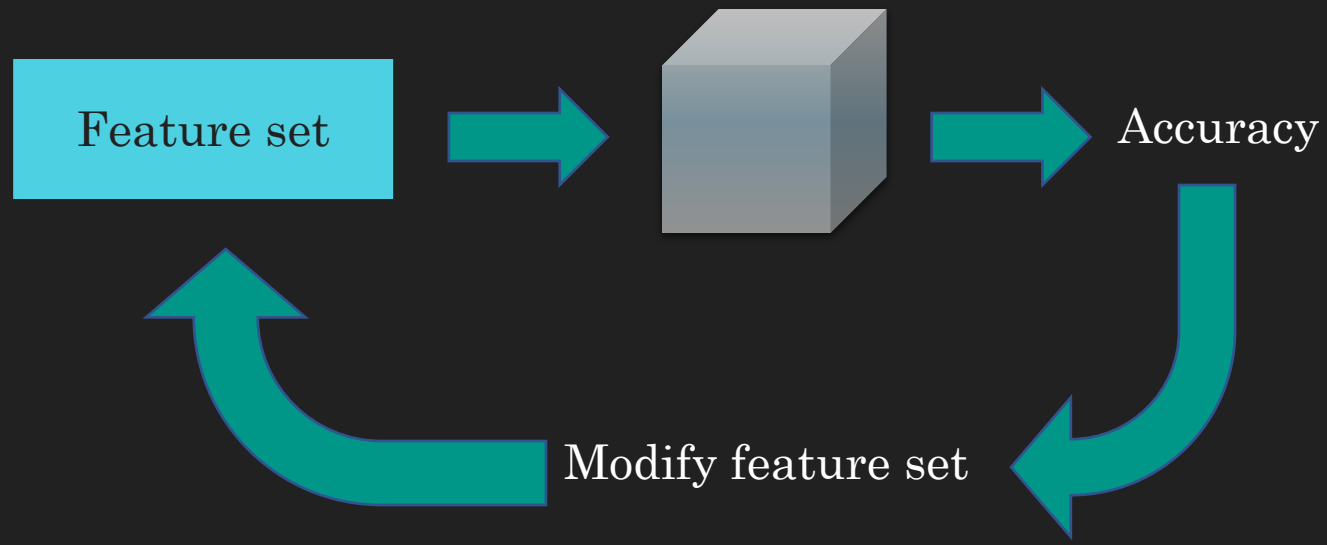
OR

(relevance / redundancy)

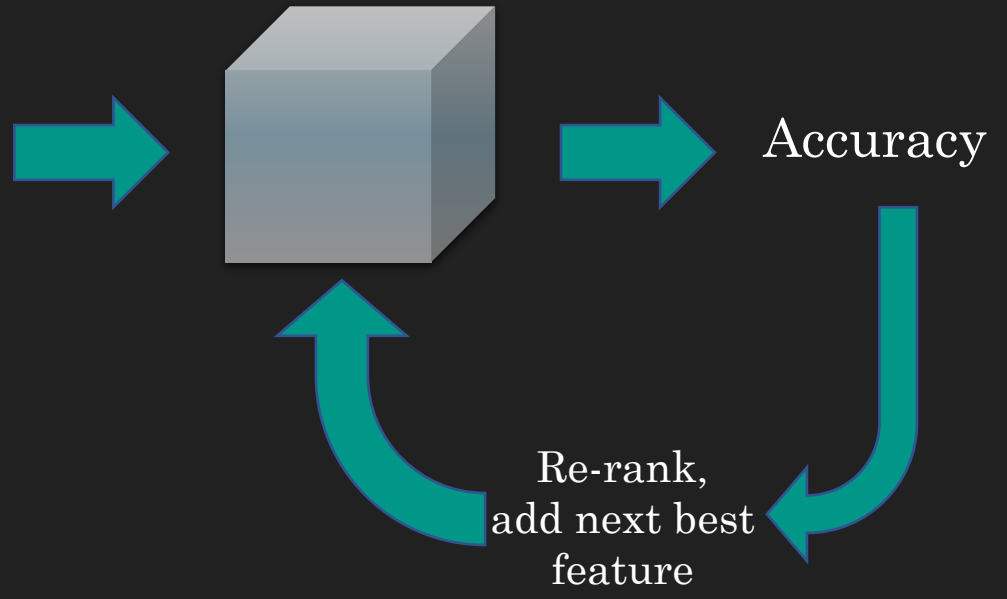
Using a greedy approach.

Wrapper Methods

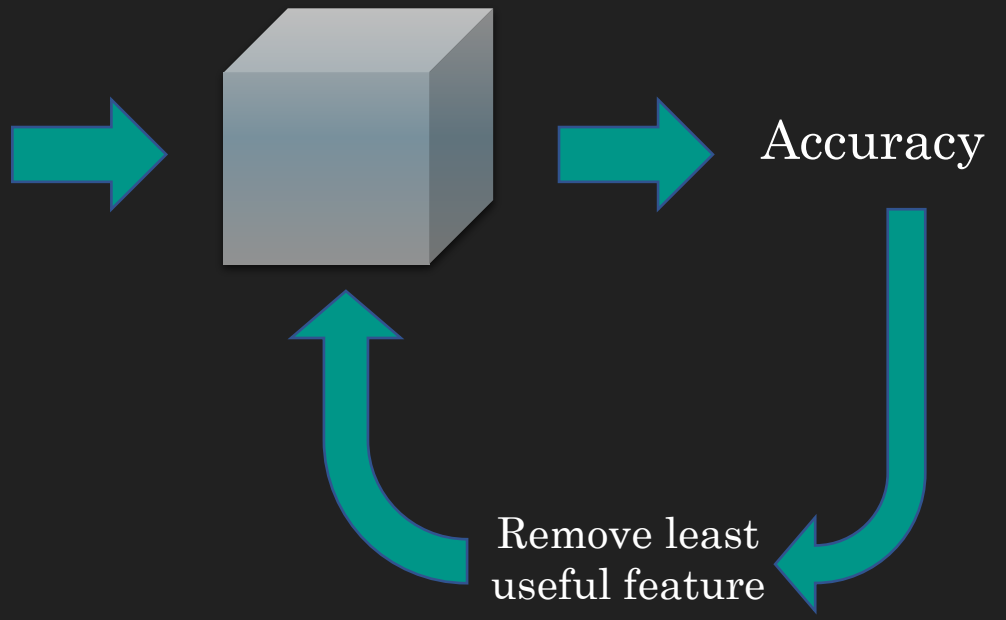
Same idea as filter methods, but instead of having a quick screening process, feedback from the full classifier is used to select features



Recursive feature addition



Recursive feature elimination

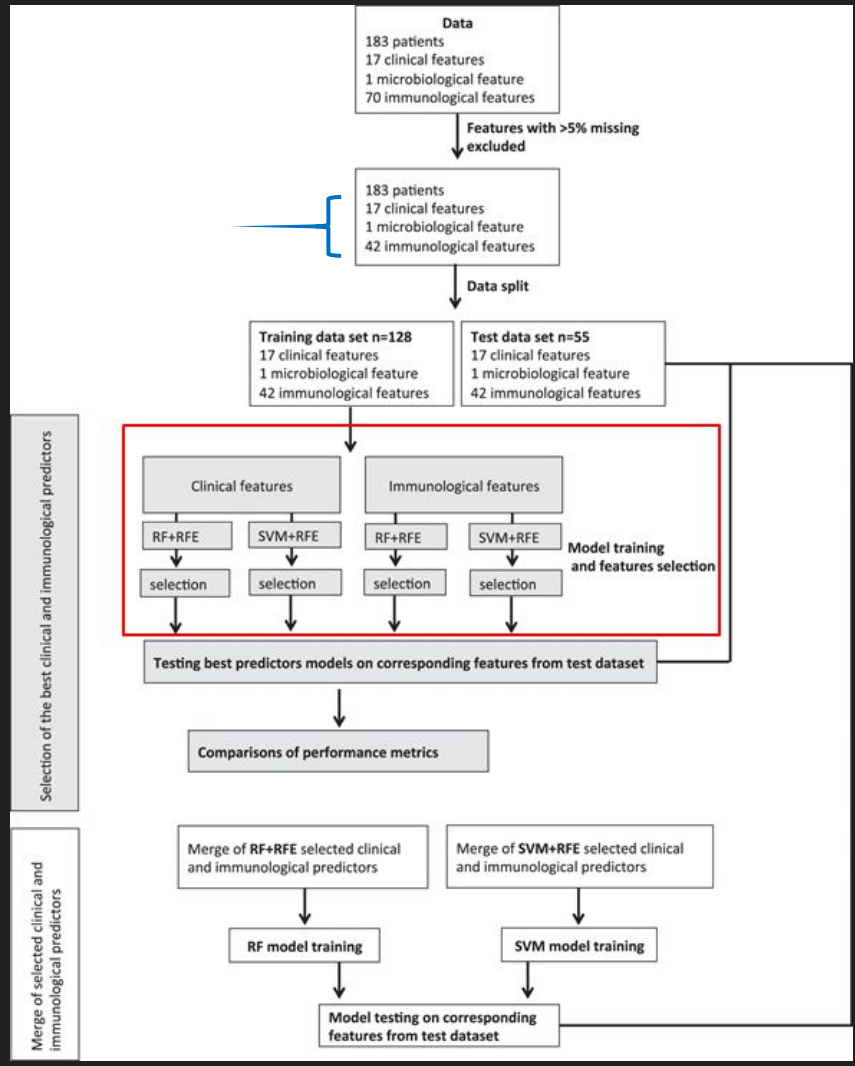


Example: recursive feature elimination

Three different definitions of urinary tract infections

What factors are the best predictors of UTI?

Try recursive feature elimination and two different classifiers: Random Forest and Support Vector Machine



Each combination of classifier and UTI definition gave a different optimal feature set

But some features (e.g., NGAL and MMP9) came up in most selected sets

Overall accuracy not great

Classifier

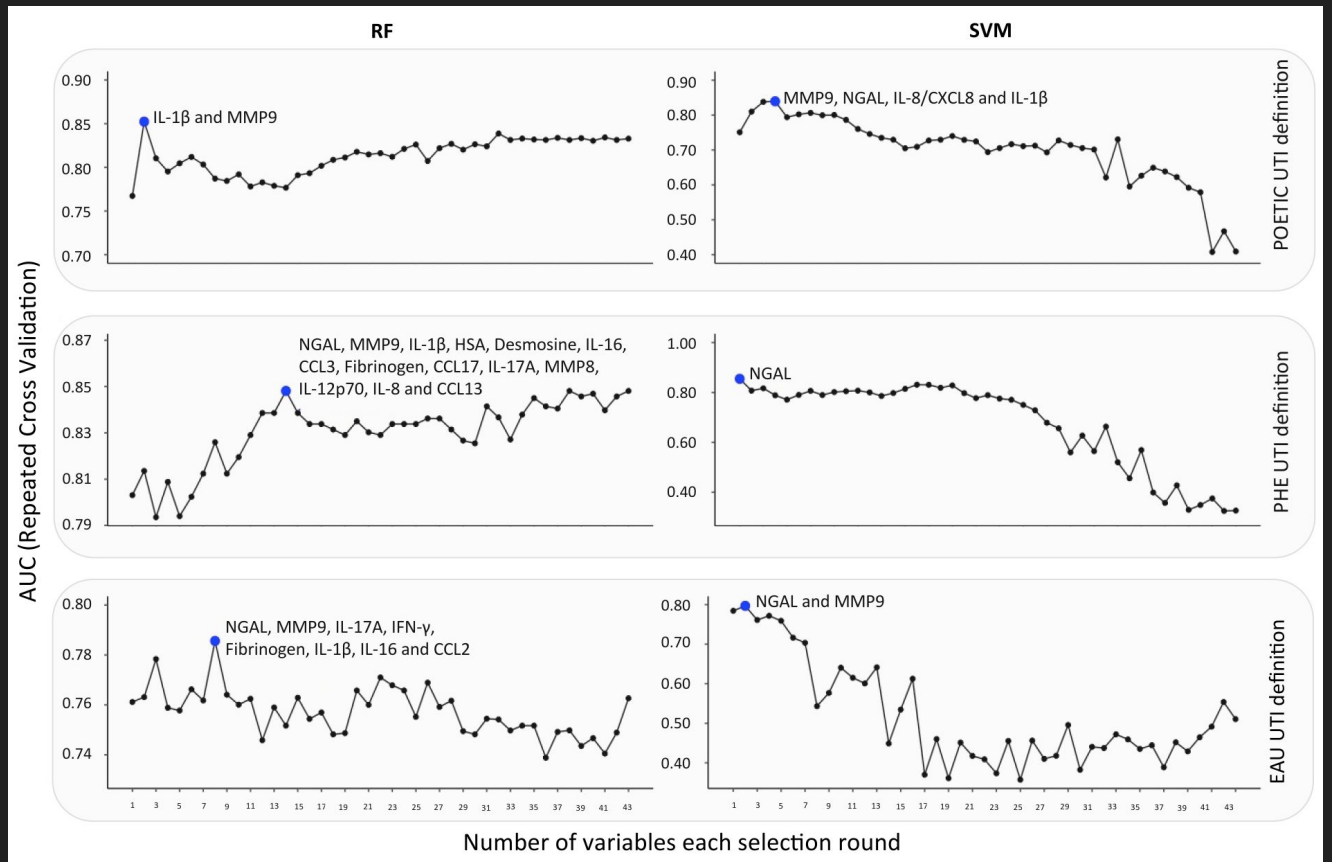


Figure S2: Feature selection among immunological markers using different UTI classification guidelines. POETIC: Point of care testing for urinary tract infection in primary care, PHE: Public Health England, EAU: European Association of Urology, AUC: Area under the ROC curve, RF: Random forest and SVM: Support vector machine

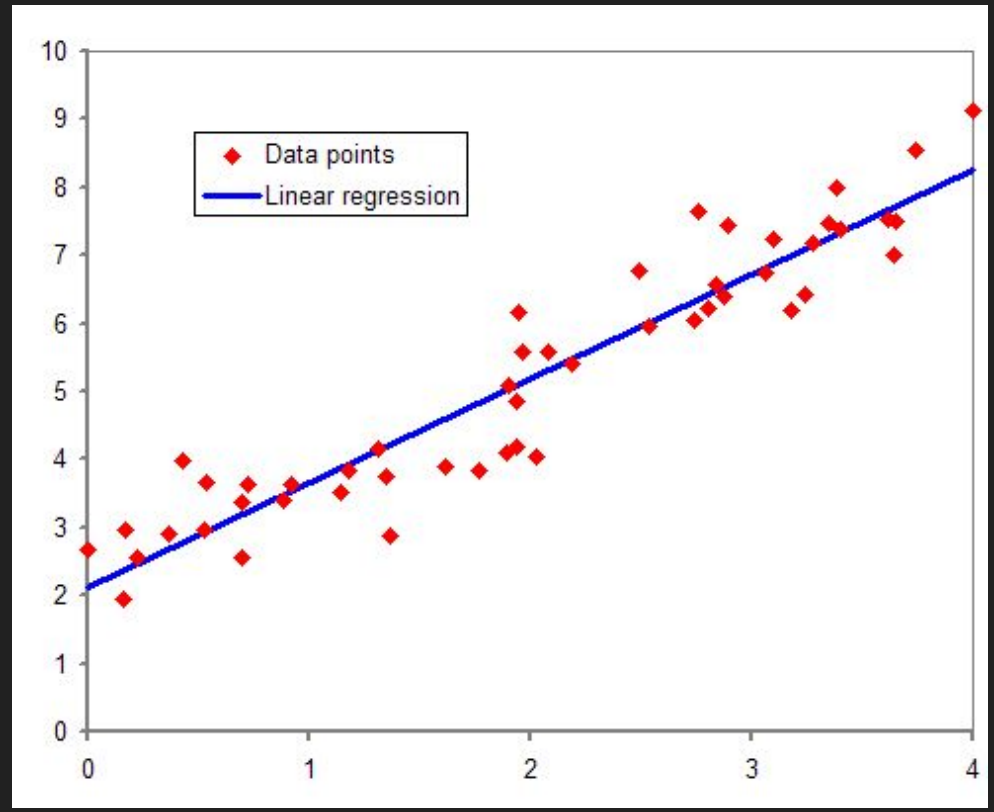
Embedded methods

Optimize feature set during model training by assigning weights

Let's think about univariate linear regression:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

- β_0 is the *y*-intercept
- β_1 is the weighting of *x* that gives the best-fit line



Multiple regression

General form:

$$y = \beta_0 + \beta_1 x_1 + \ \ldots \ + \beta_p x_p + \varepsilon$$

Predicted value

Coefficient (β_1) times value of feature 1

Error term

So many possible predictors!

- Plenty of opportunities to overfit
- Spurious relationships make it hard to interpret coefficients

Reminder: Regularization

Penalize model complexity using the λ parameter

$$Score = Accuracy - \lambda \times Complexity$$

- We can use λ in regression to minimize the influence of features on the final model

One way to regularize

- LASSO: aggressively prune features based on absolute size

$$\min_{\beta} ||y - X\beta||_2^2 + \lambda||\beta||_1$$

- Linear penalty aggressively sets many coefficients to zero (equivalent to removing features)

  Large λ: big penalty, fewer features

Another way to regularize

- Ridge regression: aggressively prune features based on SS

$$\min_{\beta} ||y - X\beta||_2^2 + \lambda ||\beta||_2^2$$

- Squared penalty is less aggressive (more features are retained, at least in part)

  Large λ: big penalty, smaller coefficients (but more non-zero features)

How do we choose?

Don’t choose! Use ElasticNet:

$$\min_{\beta} \left| \left| y - X\beta \right| \right|_2^2 + \lambda \left[\alpha ||\beta||_1 + (1 - \alpha) ||\beta||_2^2 \right]$$

There’s our friend λ

We use α ($0 \leq \alpha \leq 1$) to balance ridge and lasso components

- α small: Ridge regression dominates
- α large: Lasso dominates

Cross-validation can be used to find a suitable value for α

Example: predictive value of gene expression in cancer

- RNA-seq: sequence a random sample of the RNA expressed inside a set of cells (remember: sequencing depth matters!)

  Compare expression levels between two categories of subjects (e.g., cancer vs. control)

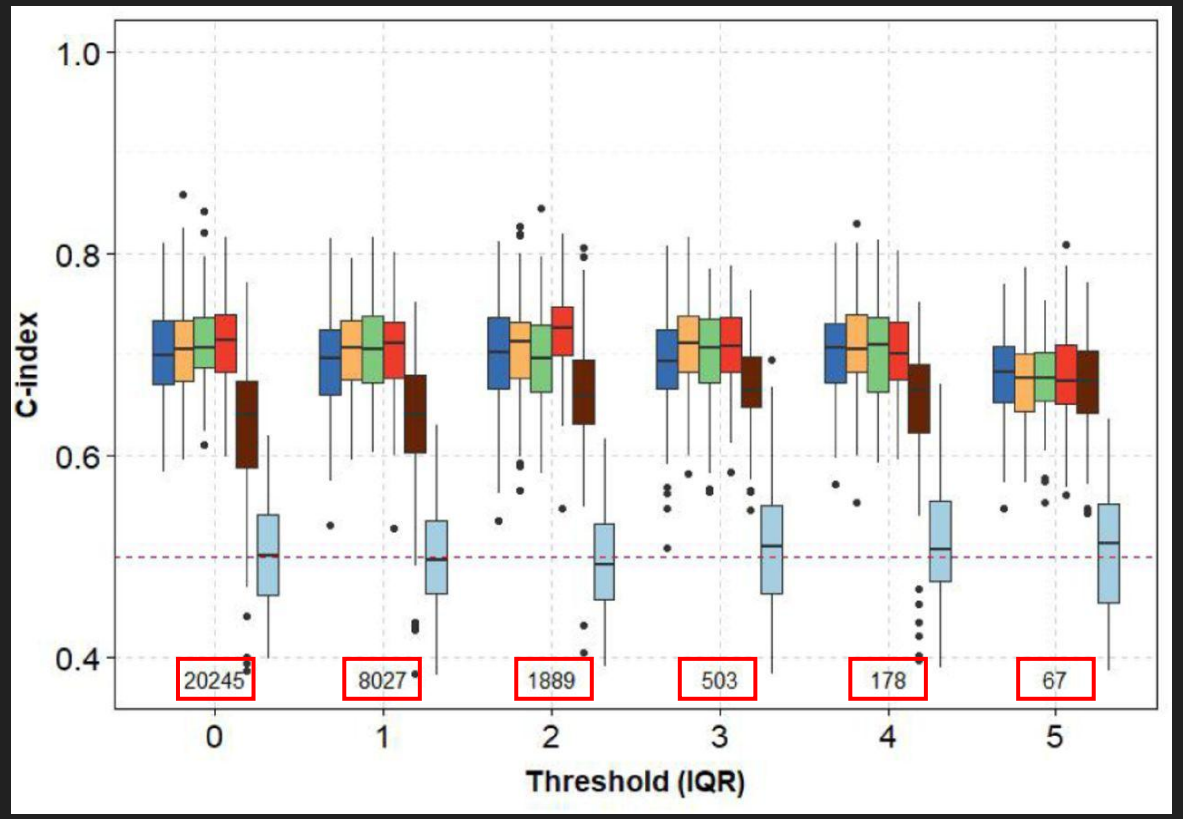
  Try out various combinations of Lasso + Ridge

  How well do the trained models perform?

  How many genes are retained?

Performance of different regression models

C-index: a measure of concordance between predictions (more correlation than plain accuracy)



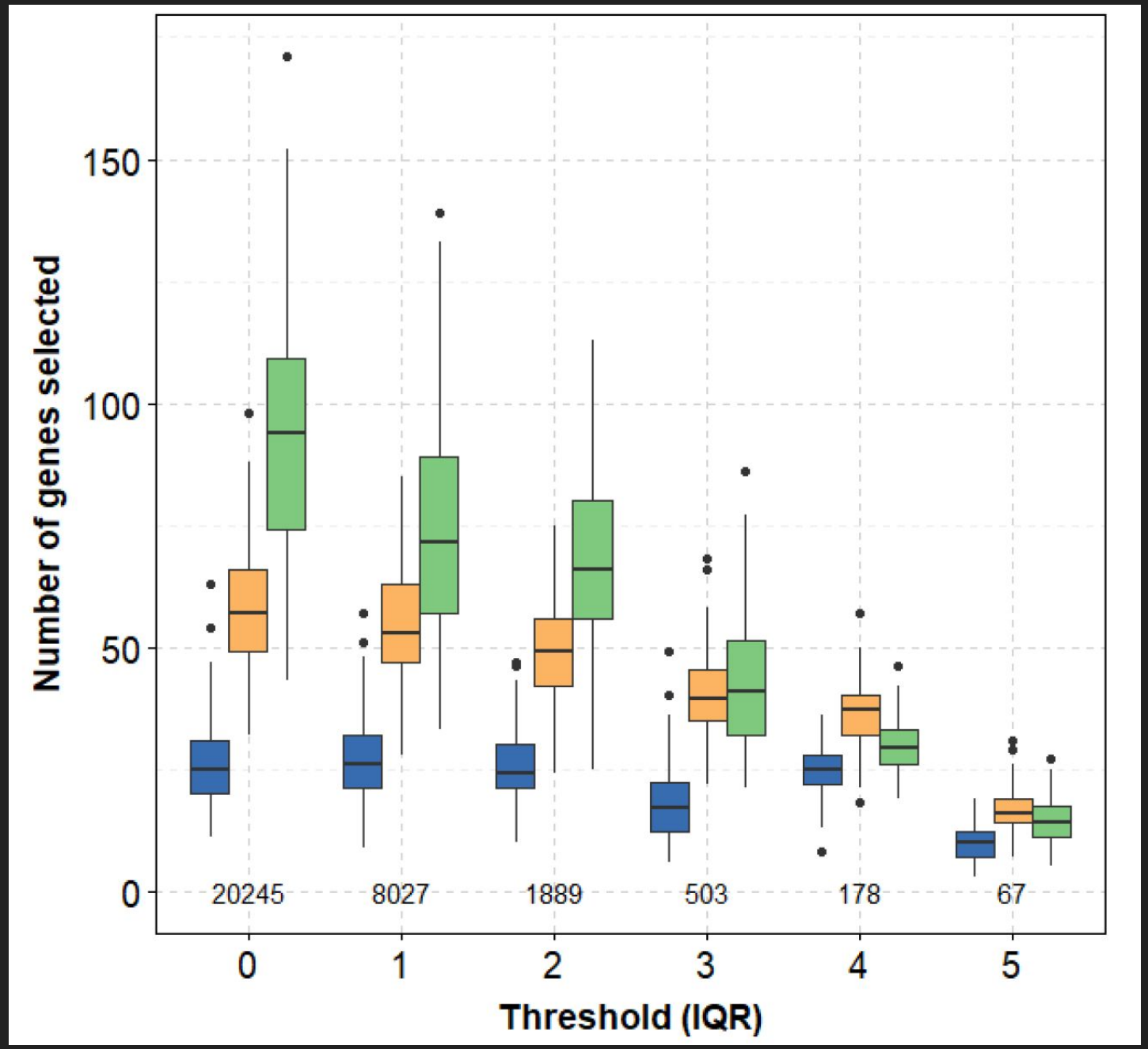
Interquartile range - simple filtering method

# of genes

Lasso
ElasticNet
Adaptive ElasticNet
Ridge
Plain regression (no penalty term)

Renal carcinoma – feature selection

- Lasso
- ElasticNet
- Adaptive ElasticNet



Feature Extraction

Try to condense *n* features into < *n* derived features or ‘metavariables’ based on correlation / covariance

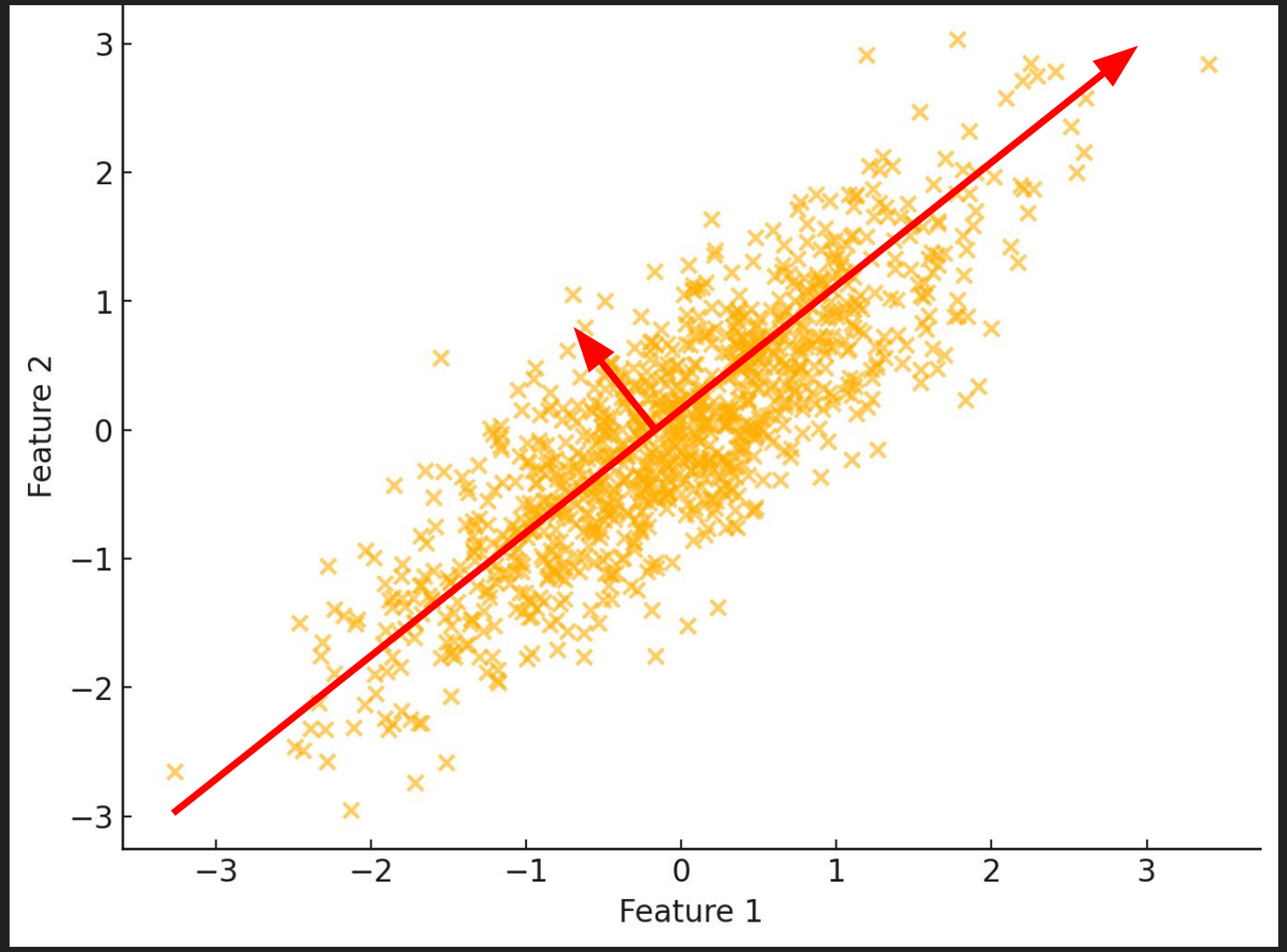
Simple example: remove 1 of 2 identical features from a data set

Principal Components Analysis

Assume that there is some degree of redundancy among features in the data set

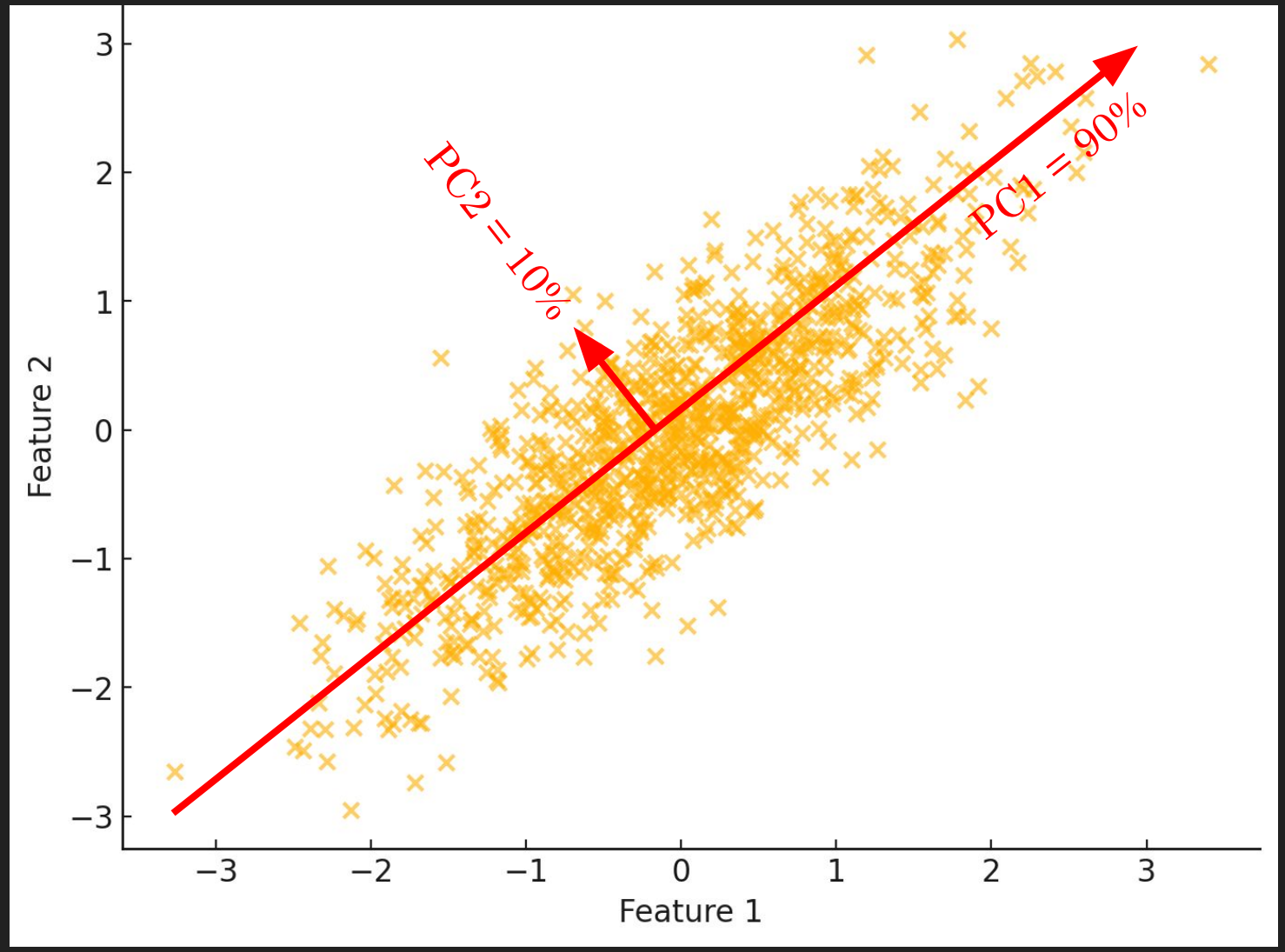
Matrix decomposition creates metavariables that capture as much of this redundancy as possible

Creation of metavariables (2D version)



- Feature 1 and 2 have some amount of covariance
- Metavariable 1 maximizes this covariance
- Metavariable 2 captures the remainder

Principal Components



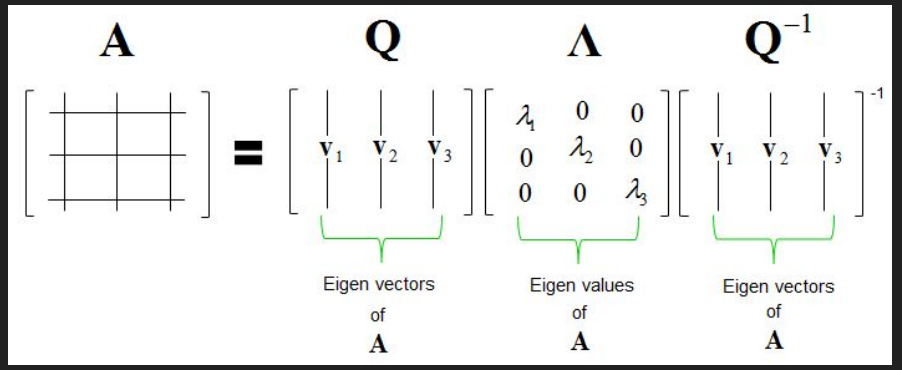
These principal components are the new features

Each explains a specific amount of the variation that was in the original features

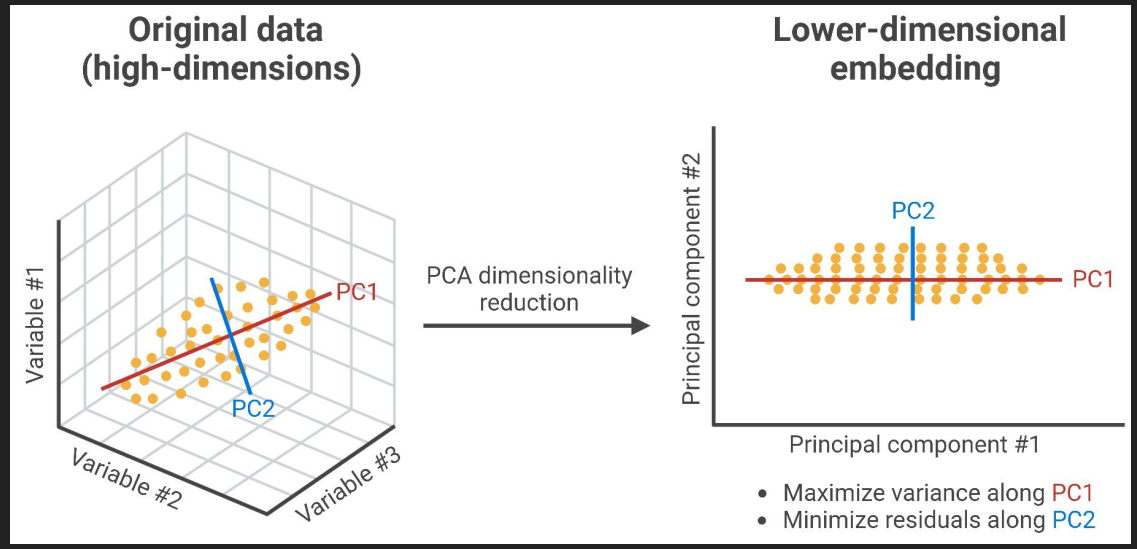
The effect is a rotation of the original coordinate system

Principal Component Analysis - Simplest Method

Reorient the data in the direction of maximal variance



1. Center the data
2. Calculate the covariance matrix
3. Perform eigendecomposition
4. Sort and select n principal components
5. Project the data onto the reduced space



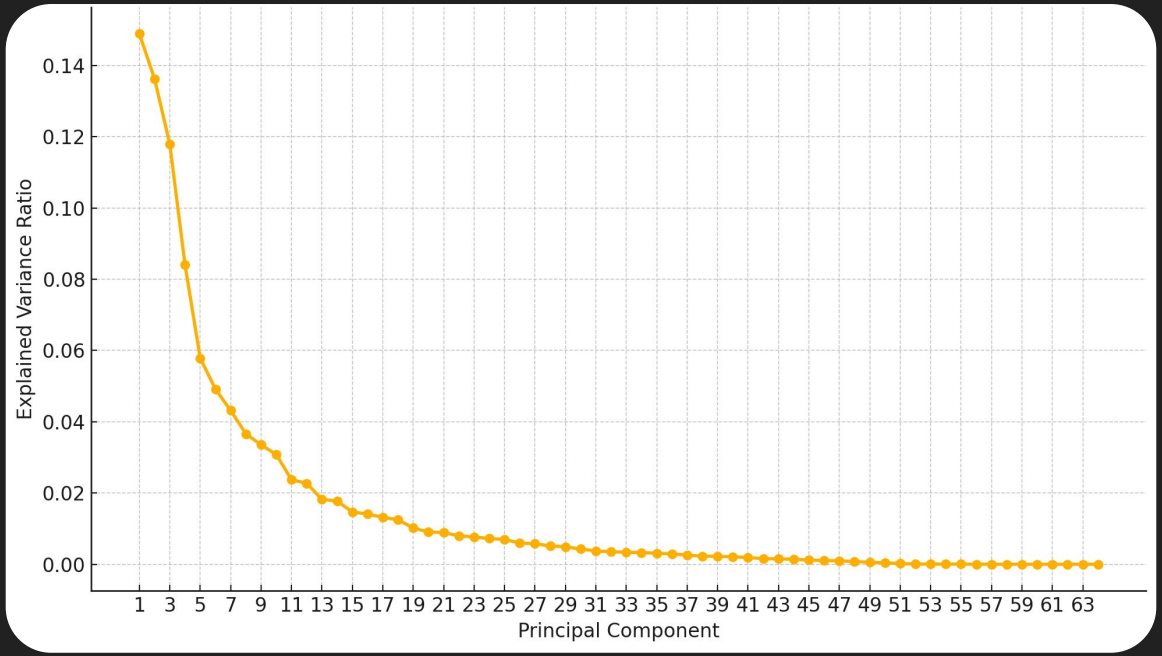
https://www.biorender.com/template/principal-component-analysis-pca-transformation

PCA gives us the same number of features!

It sure does!

But features are sorted based on the amount of variance they explain in the original data set

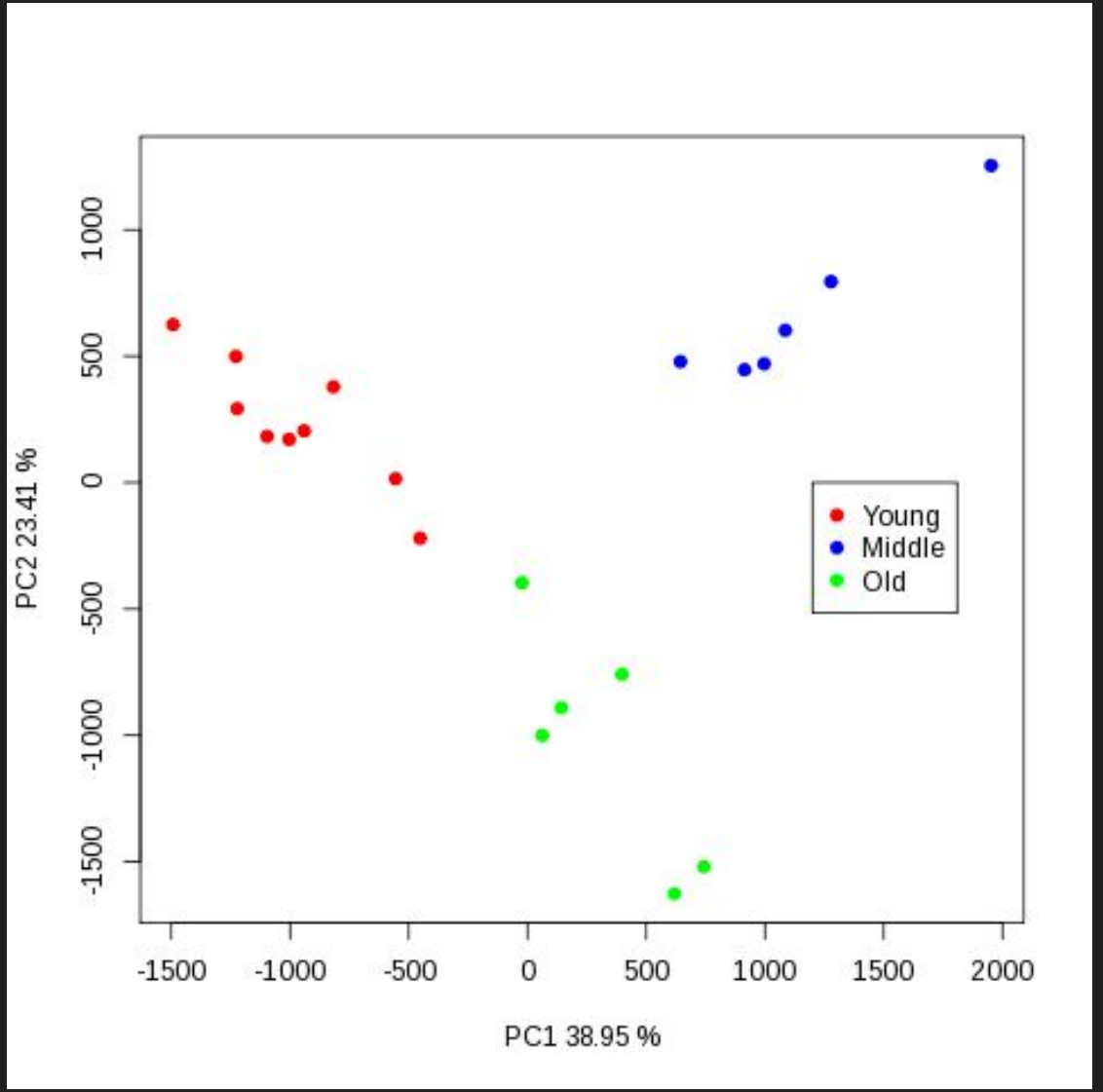
So choose only the first *k* features and discard the rest



Graphical view

Input: estimated bacterial species frequencies for 21 mouse fecal samples

Plot of first two principal components (with % of variance explained)

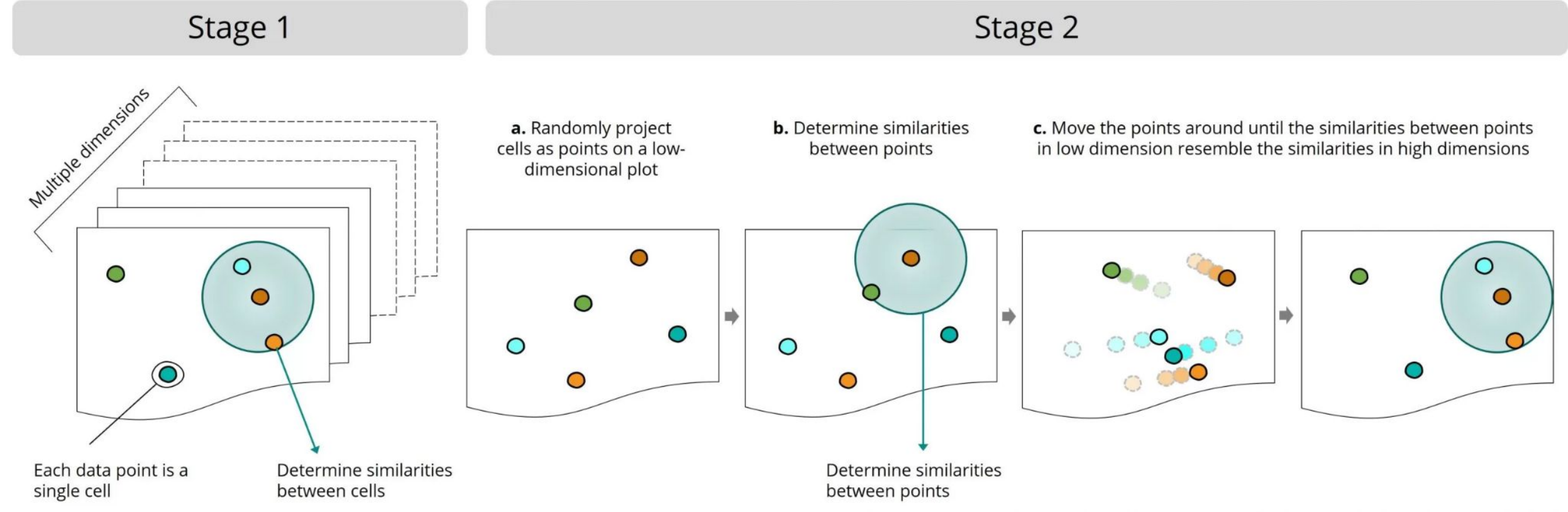


Other Feature Extraction Methods

- Principal Coordinates Analysis: Like PCA, but generated from a distance matrix calculated from the original features

- Non-parametric methods: Multidimensional scaling, t-SNE. UMAP

t-SNE (stochastic neighbour embedding) and UMAP



- Pairwise probability distribution in all dimensions
- Pairwise probability distribution in few dimensions
- Stochastic minimisation of KL divergence between distributions

https://www.scdiscoveries.com/blog/knowledge/what-is-t-sne-plot/

"With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." - von Neumann

Inadvisable for feature generation

Other Feature Extraction Methods

- Principal Coordinates Analysis: Like PCA, but generated from a distance matrix calculated from the original features
- Non-parametric methods: Multidimensional scaling, t-SNE. UMAP
- Autoencoders: neural network-based mapping of features into a lower-dimensional “latent space”

Autoencoder: find reduction by reconstructing original data

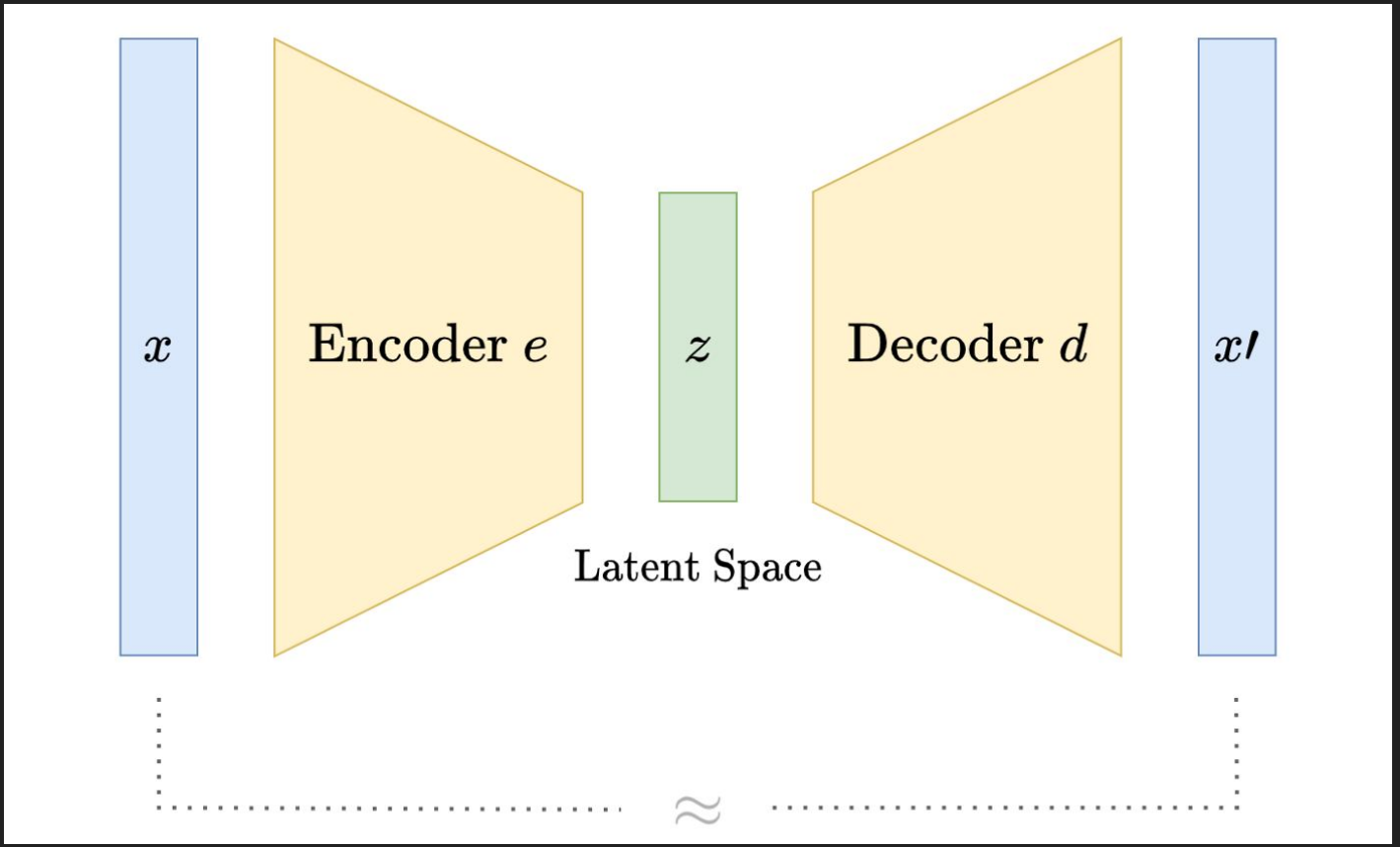
For each observation in dataset:

Find parameters:

e that compress **x** to dim(**z**)

d that recover **x** from **z**

Loss: difference between input **x** and reconstructed output **x'**



"Self-supervised"

Summary

We can generate as many features as we want from DNA and protein sequences

Not all of these will be useful or independent predictors

We should therefore reduce the complexity of the problem using good design and reduction methods