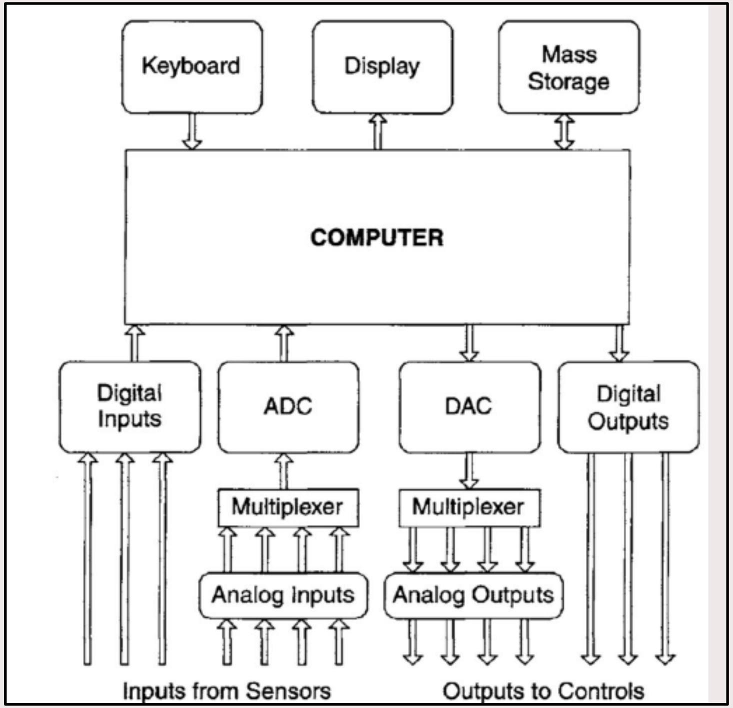
Analysis I: Feature selection & extraction

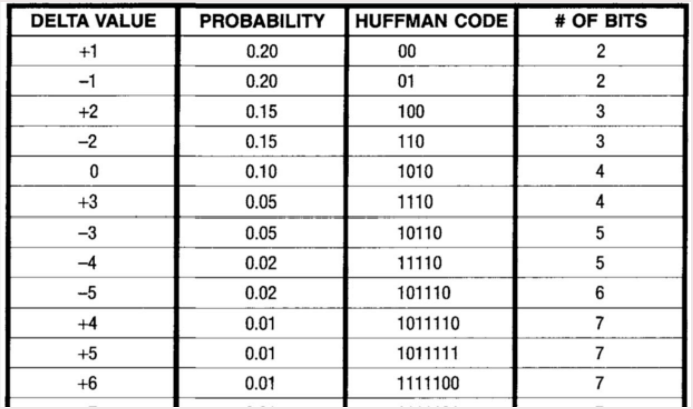
5ARB0: DATA ACQUISITION & ANALYSIS

Wouter Kouw

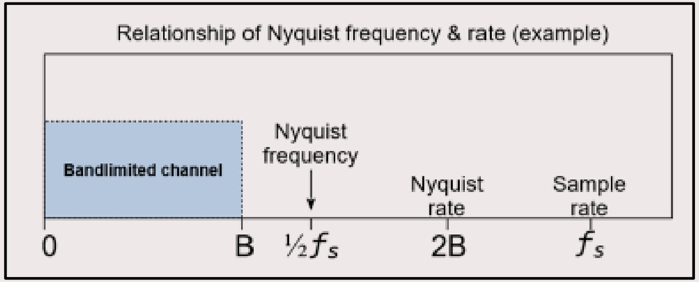
Dept EE – Signal Processing Systems group

Recap: data acquisition





DELTA VALUE	PROBABILITY	HUFFMAN CODE	# OF BITS
+1	0.20	00	2
–1	0.20	01	2
+2	0.15	100	3
–2	0.15	110	3
0	0.10	1010	4
+3	0.05	1110	4
–3	0.05	10110	5
–4	0.02	11110	5
–5	0.02	101110	6
+4	0.01	1011110	7
+5	0.01	1011111	7
+6	0.01	1111100	7



Outline

Feature selection

- What to measure?
- Selection criteria.
- Search algorithms.

Feature extraction

- Windowed characteristics.
- Principal Component Analysis.
- Independent Component Analysis.

What to measure?

Suppose you've built a drone and would like to know its airspeed in flight.

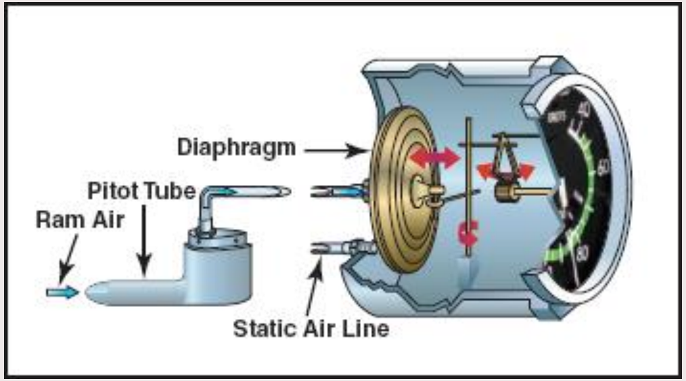
What kind of sensor would you use to measure that?

A. An atomic clock.

B. A laser distance sensor pointed at the ground.

C. A rotation sensor on the propellors.

[✓ D. An air pressure sensor.]



What to measure?

It is clear that in this case you would only need 1 sensor.

But this is because you know there exists a functional relationship between physical quantity X (air pressure difference) and physical quantity Y (airspeed).

That might not always be the case. Sometimes …

1. You do not know how to calculate Y from X.

2. You do not know whether a given X relates to Y at all.

3. Your measurement of X is too corrupted by noise.

Measuring features

A common approach is to measure many “features” of a quantity / object of interest.

- **Seismometers (geophones) spread across a volcano.**
- **Temperature and vibrations in trains.**
- **Gene expression in cancer patients.**
- **Sound length, power and filter matches for speech recognition from audio.**
- **Topological features of black strokes for optical character recognition.**

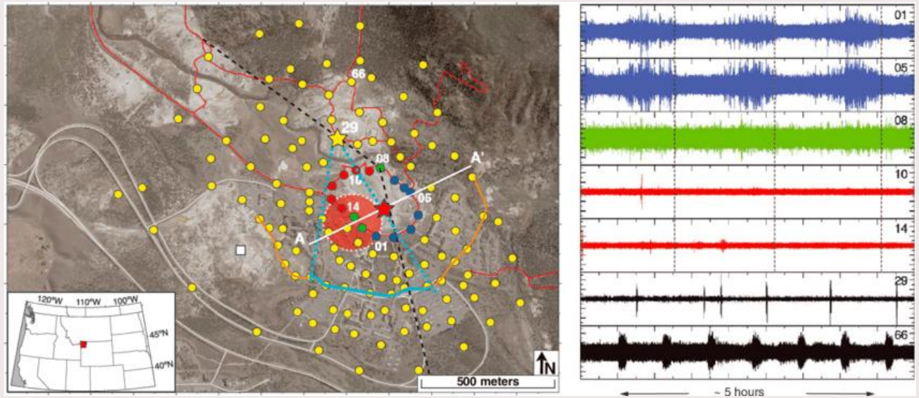


Figure taken from *Wu, Ward, Farrell, Lin, Karplus & Smith (2017), Anatomy of Old Faithful From Subsurface Seismic Imaging of the Yellowstone Upper Geyser Basin.*

Measuring features

But having lots of features comes at a cost.

- **It takes more work to train models on data.**
    - More calculations means longer training time and more power expended.
- **Models become harder to interpret.**
    - Are there many causal relationships or is there one underlying cause?
- **Some models behave unintuitively in high-dimensional spaces.**
    - Curse of dimensionality -> numerical instabilities.

Ideally, we would look for an optimal subset of features.

Selection criteria

Optimizing feature selection requires a numerical criterion (i.e., an objective function):

- **Variance**
- **Correlation**
- **Mutual information**
- **Scatter-based criteria (clusters)**
- **Mean squared error (regression)**
- **Error rate / accuracy (classification)**

Selection criteria: variance

Variance **of a random variable is defined as:**

$$\mathbb{V}[\boldsymbol{x}] = \mathbb{E}\left[(\boldsymbol{x}-\boldsymbol{\mu})^2\right] = \mathbb{E}\left[\boldsymbol{x}^2\right] - \mathbb{E}[\boldsymbol{x}]^2$$

It may be estimated from data with:

$$\widehat{\mathbb{V}}[\boldsymbol{x}] = \frac{\mathbf{1}}{\boldsymbol{N}-\mathbf{1}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}}(\boldsymbol{x}-\widehat{\boldsymbol{\mu}})^{\mathbf{2}}$$

where N is the number of samples and $\widehat{\boldsymbol{\mu}} = \frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}} \boldsymbol{x}_{\boldsymbol{i}}$ is the sample average.

Variance thresholding is simply rejecting features that do not vary significantly within the timeframe of interest.

- Often used to automatically remove “dead” sensors out of an array.

Selection criteria: correlation

Covariance is the amount that two random variables vary simultaneously:

$$\mathbb{C}[\boldsymbol{x}, \boldsymbol{y}] = \mathbb{E}[(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])(\boldsymbol{y} - \mathbb{E}[\boldsymbol{y}])]$$

*Correlation** is covariance normalized by the standard deviations of the variables:

$$\boldsymbol{r}(\boldsymbol{x}, \boldsymbol{y}) = \frac{\mathbb{C}[\boldsymbol{x}, \boldsymbol{y}]}{\boldsymbol{\sigma}_{\boldsymbol{x}} \boldsymbol{\sigma}_{\boldsymbol{y}}}$$

where standard deviations are the square roots of variances:

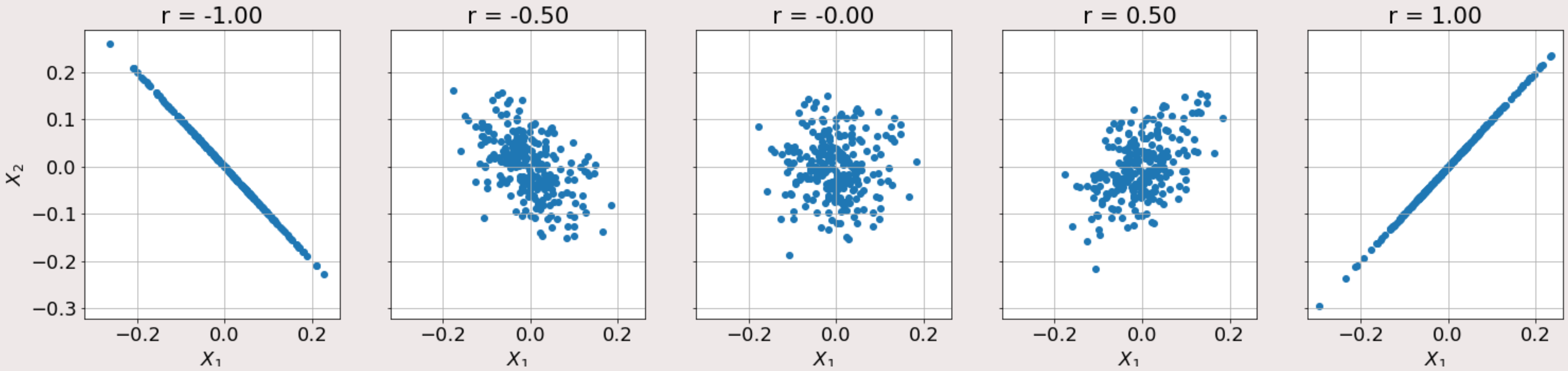
$$\boldsymbol{\sigma}_{\boldsymbol{x}} = \sqrt{\mathbb{V}[\boldsymbol{x}]} \text{ and } \boldsymbol{\sigma}_{\boldsymbol{y}} = \sqrt{\mathbb{V}[\boldsymbol{y}]}$$

Covariance – and thereby correlation - can be estimated in a similar manner as variance.

*Pearson’s correlation coefficient

Selection criteria: correlation

Correlation coefficients range within [-1, +1]:



If two variables are perfectly correlated, then one can be recovered from the other.

One can select features based on pairs with the least correlation between them.

Selection criteria: mutual information

Mutual information (MI) is intuitively similar to correlation but is more general.

To go through its definition, we must first discuss entropy.

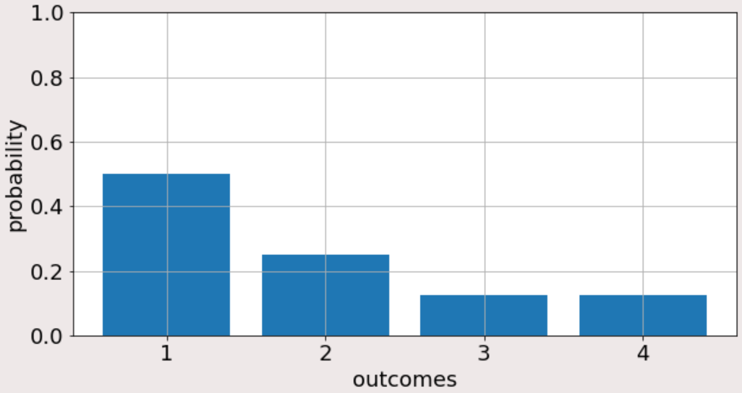
Q: What is the Shannon entropy of the random variable with the probability mass below?

[✓ A. $H[X] = 1\frac{3}{4}$]

B. $H[X] = 8$

C. $H[X] = \frac{1}{4}$

D. $H[X] = 9$



Selection criteria: mutual information

(Shannon) entropy is defined as the expected log-probability of a random variable X:

$$H[X] = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$$

where $p(x) = \Pr[X = x]$ and $\mathcal{X}$ is the event space.

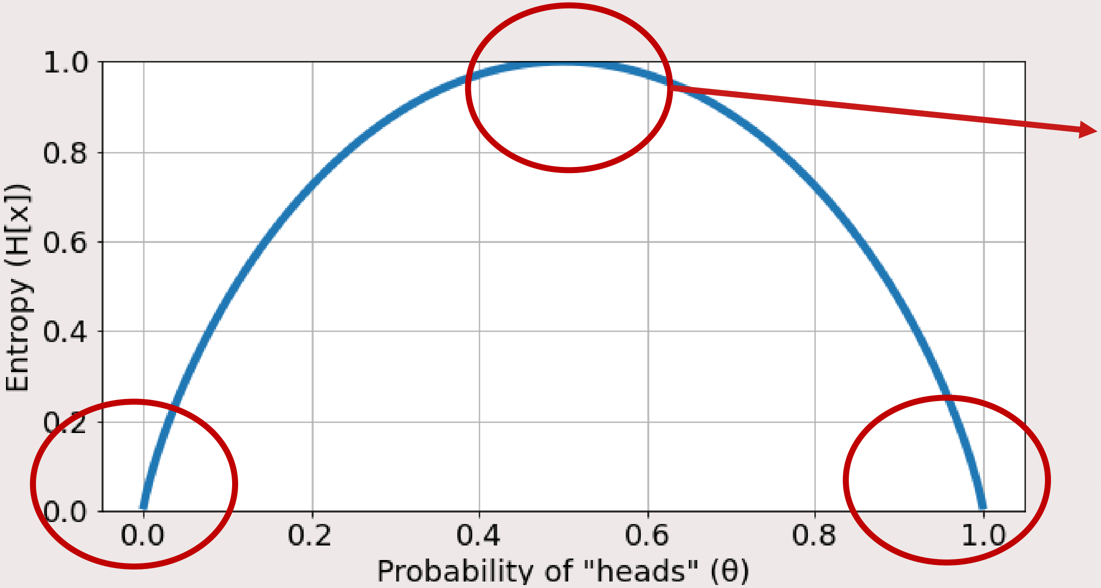
Originally put forward by Claude Shannon to characterize information content communicated between two parties.

- Core idea: rare events are *surprising* / more *informative.*

Entropy can be estimated by computing the relative occurrence of each event (i.e., a histogram) and evaluating the formula above (note that $0 \log 0 = 0$).

Selection criteria: mutual information

Example: suppose you throw an unfair coin with probability θ of "heads":



If both outcomes are equally likely, then each event is somewhat surprising and the average surprise is maximal.

If most of the outcomes are "tails", then most outcomes are not *surprising* and the average *surprise* is small.

Same situation for most outcomes are "heads".

Selection criteria: mutual information

The generalization to continuous variables is called *differential entropy:*

$$\boldsymbol{H[X] = -\int_{x} p(x)\log p(x)\,dx}$$

It is an analogy to entropy for discrete random variables:

- It is not a limit of Shannon entropy for the number of events going to infinity.
- It can have negative values (hard to interpret).

It can be estimated by fitting a parametric probability distribution (e.g., Gaussian) to the data and then taking the expectation with respect to the log pdf.

- If you fit a wrongly assumed distribution, you will get an approximation error.

Selection criteria: mutual information

Now we move on to *conditional entropy*:

$$\boldsymbol{H[X|Y] = \sum_{y \in \mathcal{Y}} p(y) H[X|Y = y]}$$

$$\boldsymbol{= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(y)}}$$

where X, Y are discrete random variables and $p(x, y)$ is a joint distribution.

It should be interpreted as the remaining randomness in X given Y.

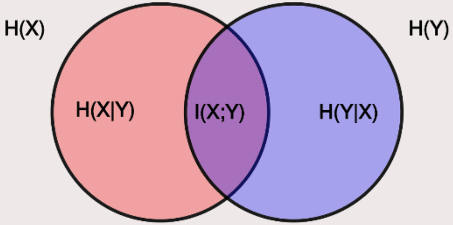
- “How surprising is an outcome $\mathrm{X} = x$ given that we have already observed $Y = y$.”

It can be estimated in the same way as Shannon entropy (i.e., with histograms).

Selection criteria: mutual information

The mutual information between random variables X, Y is:

$$
\begin{aligned}
I[X;Y] &= H[X] - H[X|Y] \\
&= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(y)p(x)}
\end{aligned}
$$



It captures the information gain achieved by jointly observing X and Y.

- If $\mathbf{X},\mathbf{Y}$ are independent, then no information can be gained by joint observation.

For Gaussian distributions, MI is directly related to correlation: $I = \frac{1}{2}\log(1 - r^2)$.

But unlike correlation, MI can also characterize information gain between non-Gaussians.

Selection criteria: mutual information

Mutual information is a natural feature selection criterion.

- The larger the value of $\boldsymbol{I[X, Y]}$, the more redundant the features are.
- We can look for the most *independent* combination of features.
- Correlation is pairwise, but MI can be used between sets of variables of any size. For example, between features {1,2,3} and {4,5}: $\boldsymbol{I[X_1, X_2, X_3; X_4, X_5]}$**.**

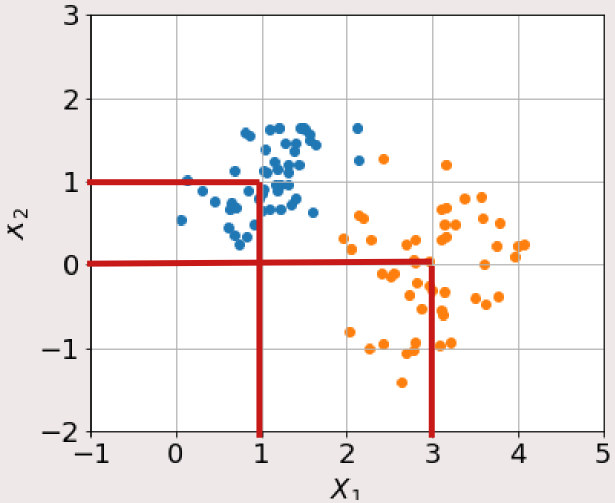
MI can also be used in conjunction with classification and regression tasks.

- We would compute the mutual information between the features and the target variable (i.e., labels or response).

Selection criteria: scatter

In cases where data has been clustered or labelled, we want to maintain separability.

- If we select X_2, then the cluster means lie a distance of 1 away from each other.
- If we select X_1, then the cluster means lie a distance of 2 away from each other.



One can assume that both clusters are Gaussian distributed, but that is not strictly necessary.

Selection criteria: scatter

Let us define the following scatter matrices on a data set of N observations:

Within-scatter:

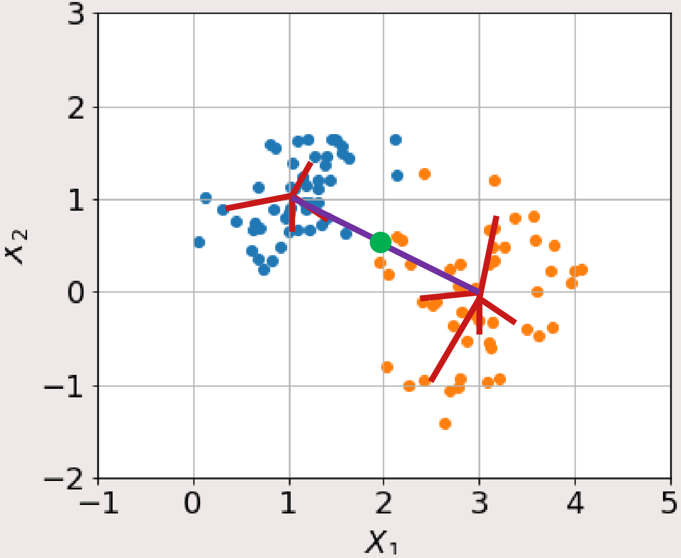
$$S_W = \sum_{k=1}^{K} \frac{N_k}{N} \Sigma_k$$

where Σ_k is the covariance matrix of the k-th class.

Between-class scatter:

$$S_B = \sum_{k=1}^{K} \frac{N_k}{N} (m_k - m)(m_k - m)^T$$

where m_k is the mean of the k-th class and m is the global mean.



Selection criteria: MSE / Error rate

In regression tasks, we can use the performance of the regression model (typically Mean Squared Error) on a validation data set as a selection criterion.

In classification tasks, we can use the classification error rate (1-accuracy).

One would train the model with and without a selected feature (set) and decide whether the increase in performance is worth the extra cost of training the model.

- In a later lecture, we will see “sparse” regression / classification models that incorporate the selection directly into the model training procedure.

Search algorithms

Suppose you want to select 2 features out of a set of 4 features in total.

Q: How many possible options are there?

A. There are 3 different sets.

B. There are 4 different sets.

[✓ C. There are 6 different sets.]

D. There are 8 different sets.

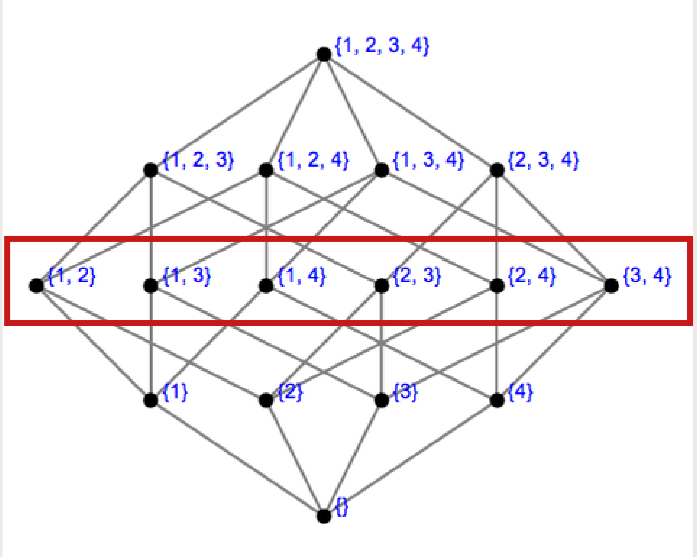


Figure taken from *Wolfram Demonstrations: Hasse Diagram of Power Sets* (link).

Search algorithms

If you do not fix the number if features to be selected, feature selection becomes a combinatorial optimization problem.

Combinatorial problems are notorious for being (prohibitively) expensive to solve exactly.

Search algorithms do not evaluate every option, but follow a designed strategy.

- **Example: A* search, genetic algorithms.**
- **Strategies provide a trade-off between optimality and computational cost.**

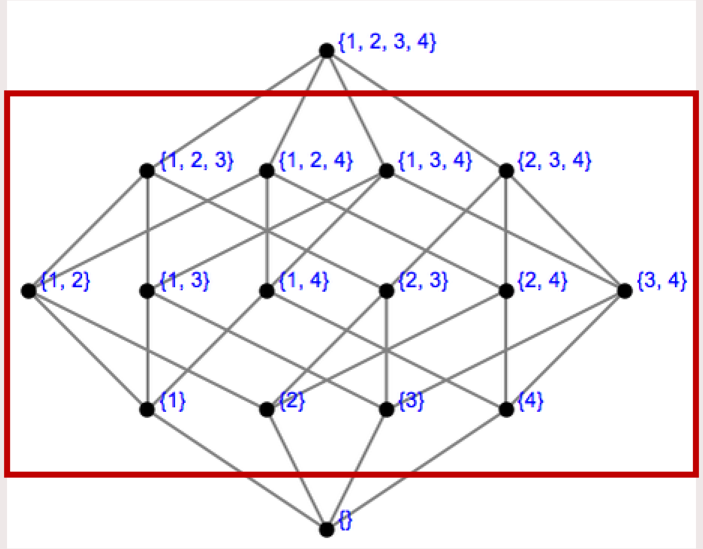


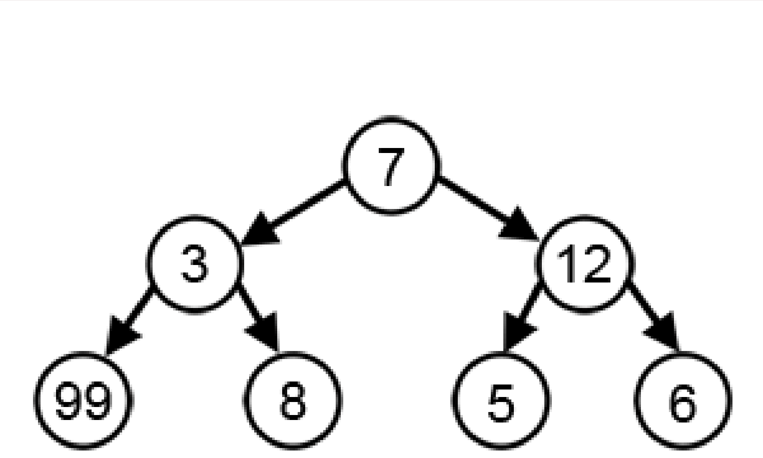
Figure taken from *Wolfram Demonstrations: Hasse Diagram of Power Sets (link).*

Search algorithms: greedy

A greedy search algorithm sequentially adds features and never drops any.

Main advantages:

- **Simple to implement.**
- **Computationally efficient.**



Disadvantages:

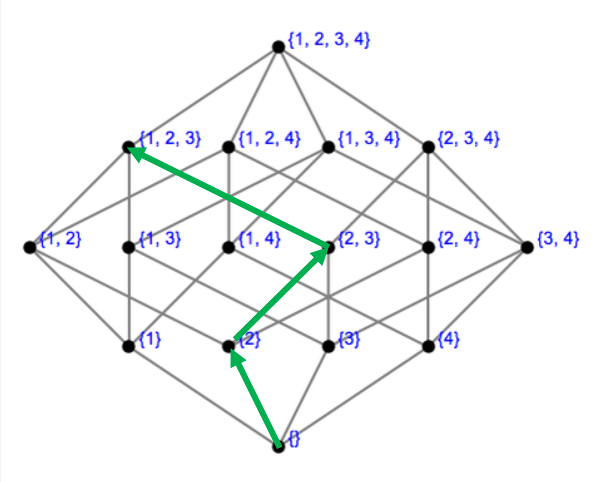
- **May lead to suboptimal solutions.**
- **Final selection is highly dependent on starting feature.**

Figure taken from *Wikipedia: Greedy algorithm* (link).

Search algorithms: elimination

Feature elimination is just greedy forward feature selection run backwards.

Forward



Backward

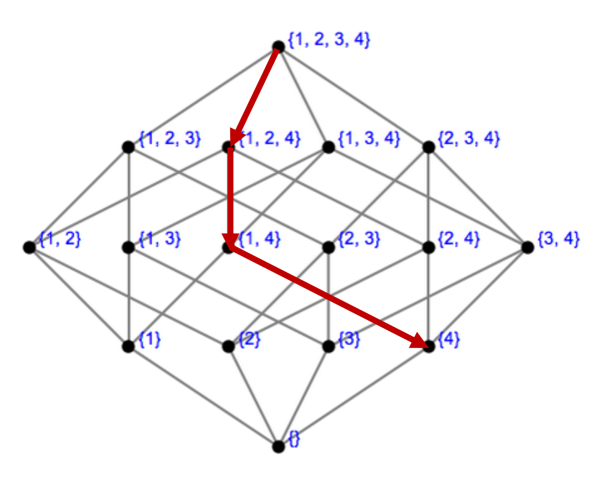
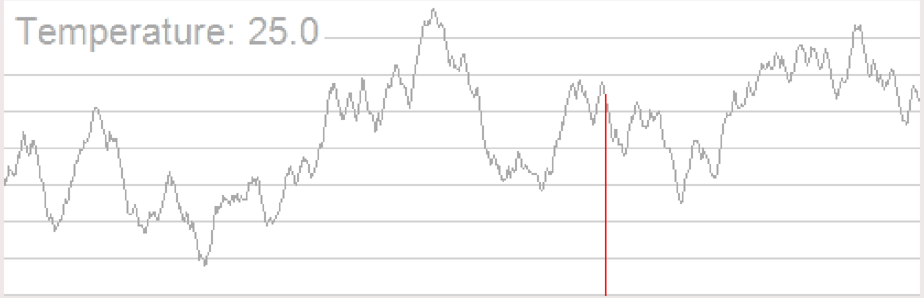


Figure taken from *Wolfram Demonstrations: Hasse Diagram of Power Sets* (*link*).

Search algorithms: simulated annealing

Annealing is the process of heating a material to a high energy state and then forcing it into a stable low energy state through a cooling protocol.

This procedure can be used for optimization as well:



For feature selection, the algorithm would initially pick random feature subsets and over time would focus on variations of the previously best evaluated subsets.

Summary: feature selection

A common approach to modeling a phenomenon whose causal relationships we don't fully understand is to collect many features of a quantity / object / event of interest.

- But having many features can be problematic, such as increasing model training time, reducing interpretability and suffering from the curse of dimensionality.

Features may be selected automatically on the basis of an optimality criterion, such as variance, correlation, mutual information or class-/cluster-based metrics.

We typically cannot do exhaustive search over the space of all feature subsets because there are simply too many.

- Search algorithms, such as greedy selection, trade off accuracy for computational cost.

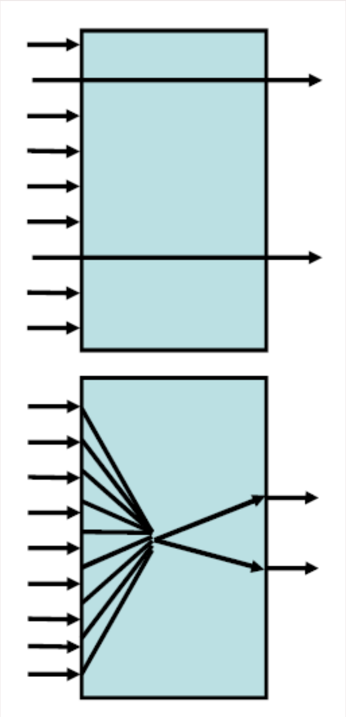
Break

Feature Extraction

In *feature selection*, features are dropped entirely.

In *feature extraction,* we create new features based on (combinations of) existing features.

- Extracting properties over windows of time.
- (Non)linear change of bases (e.g. frequency spectrum).
- Similar to data compression (e.g. JPG, MPEG, MP3).
- Combinations can be designed or learned.



Windowed characteristics

It may be appropriate to extract specific numerical properties over a given time window.

- Example: tracking the maximum heart rate per day of a patient.

You extract a new feature from one (or more) existing feature(s) through a known mapping, such as *max, min, standard deviation, mean* (similar to an averaging filter).

Mostly done if there is prior knowledge of the timescale of the events under study.

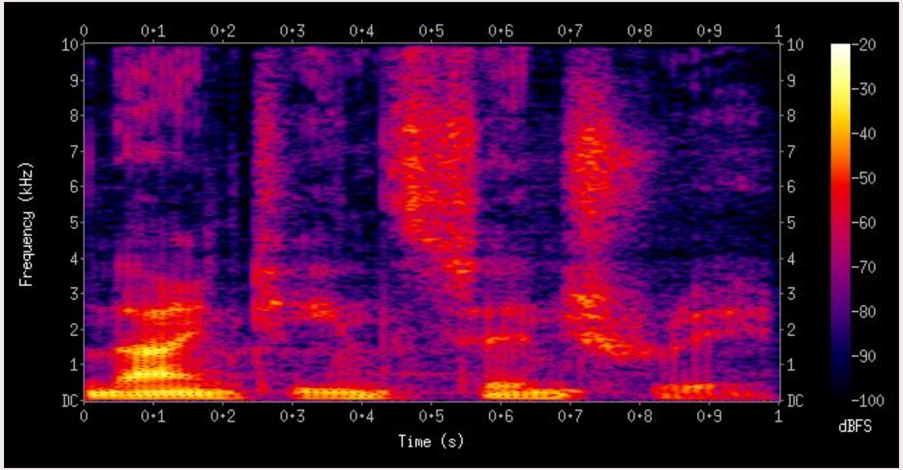
- Finding a “good” feature to extract is highly task dependent.
- One may optimize a parameterized feature (see *unsupervised learning* in next lecture).

Windowed characteristics

When you're working with signals, a natural extension is to consider spectrograms.

A spectrogram can be constructed by a Fourier transformation on a window of the signal sliding over time.

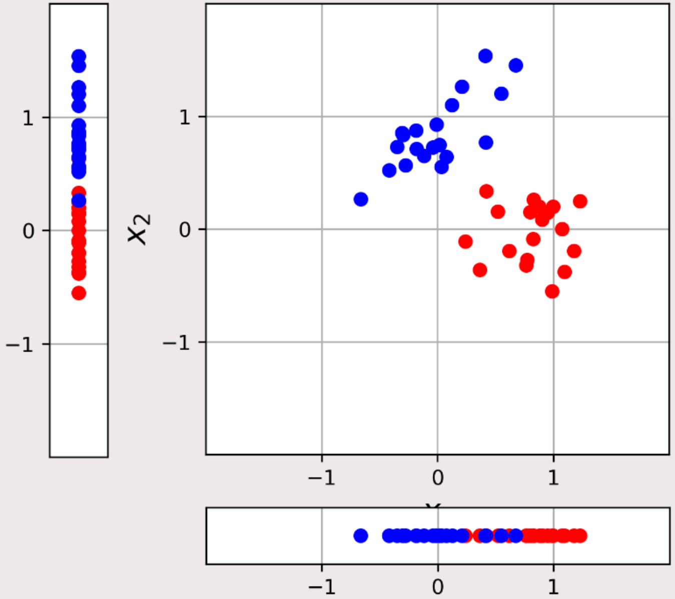
- X-axis displays time (seconds).
- Y-axis displays frequency (Hz).
- Color axis displays power (dB).



Linear changes of representation

Suppose you are given the following data set:

- Selecting $\boldsymbol{X_1}$ will cause overlap.
- Selecting $\boldsymbol{X_2}$ will also cause overlap.



Q: Can you represent the data with 1 feature such that the clusters do not overlap?

Yes: $\boldsymbol{Z = \frac{1}{2}X_1 + \frac{1}{2}X_2}$

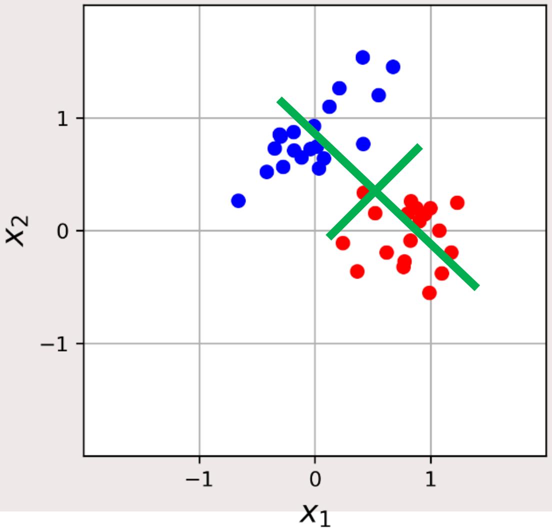
Linear changes of representation

But visual inspection is not always possible (or preferred).

Q: How would you automatically find an appropriate change of bases?

You might try to find a basis where each direction corresponds to some property of the data.

- Variance -> Principal Component Analysis.
- If the bases are orthogonal, then the extracted features will be linearly uncorrelated.



Principal Component Analysis

We aim to extract a feature $Z = Xw$ where the linear transformation w is:

$$\begin{aligned} \underset{w}{\textbf{maximize}}\ & \mathbb{V}[\boldsymbol{Xw}] \\ \boldsymbol{s.t.}\quad & \boldsymbol{w^T w = 1} \end{aligned}$$

The constraint imposes the solution vector to have unit length.

We can re-arrange the objective to:

$$\mathbb{V}[\boldsymbol{Xw}] = \boldsymbol{w^T}\mathbb{V}[\boldsymbol{X}]\boldsymbol{w} = \boldsymbol{w^T \Sigma w}$$

where $\boldsymbol{\Sigma}$ is the covariance of the data.

Principal Component Analysis

We form a Lagrangian using the objective and the equality constraint

$$\mathcal{L}(w,\lambda) = w^T \Sigma w - \lambda\left(w^T w - 1\right)$$

where λ is the Lagrange multiplier corresponding to the constraint.

Taking the gradient and setting it to 0 gives:

$$\frac{\partial}{\partial w}\mathcal{L}(w,\lambda) = \Sigma w - \lambda w \equiv 0$$

We recognize an eigendecomposition problem: w (the eigenvector) will point to the direction of maximal variance with λ (the eigenvalue) the variance in that direction.

Principal Component Analysis

But now we just have a single component $\boldsymbol{w_1}$.

Q: How do we obtain a second component $\boldsymbol{w_2}$ orthogonal to the first one?

We add constraints:

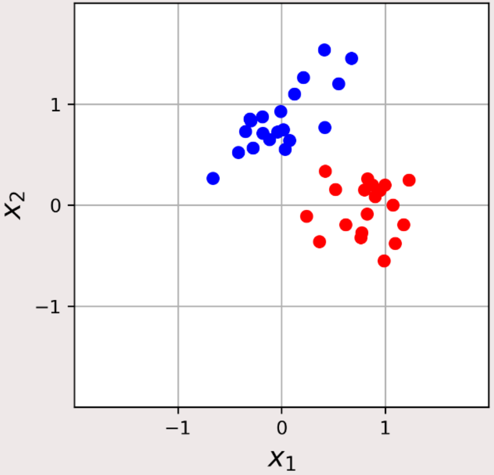
$$\begin{aligned} \underset{\boldsymbol{w_2}}{\textbf{maximize}} \quad & \mathbb{V}[\boldsymbol{Xw_2}] \\ \boldsymbol{s.t.} \quad & \boldsymbol{w_2^T w_2} = \mathbf{1} \\ & \boldsymbol{w_2^T w_1} = \mathbf{0} \end{aligned}$$

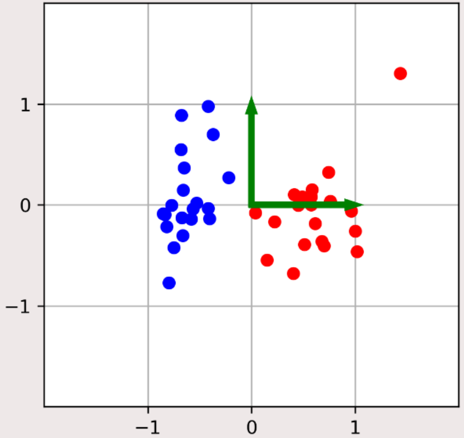
This can again be tackled by forming a Lagrangian and solving for $\boldsymbol{w_2}, \boldsymbol{\lambda}$.

The second component is the second largest eigenvector of the covariance matrix.

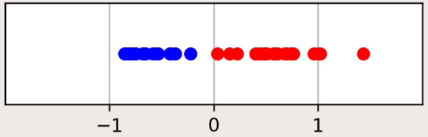
Principal Component Analysis

Applying the linear transformation on our example data set produces:





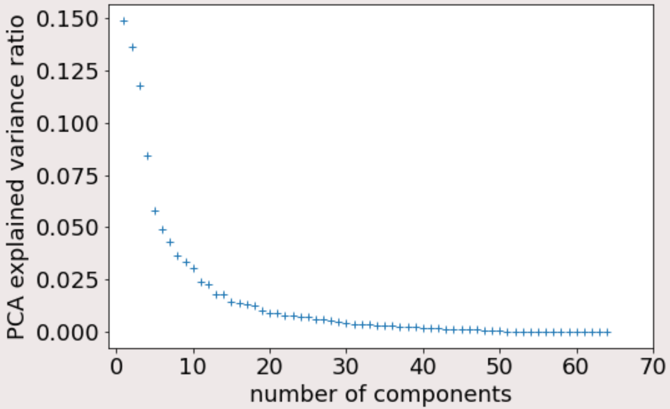
Clusters in the first feature do not overlap:

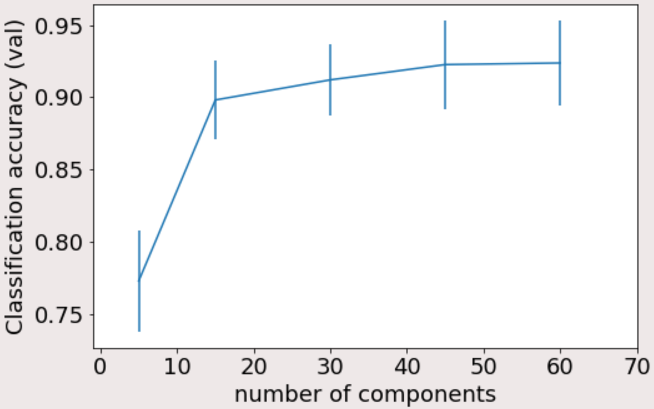


Principal Component Analysis

The optimization procedure can be repeated up to the dimensionality of the data.

- The eigenvalues drop off as you go, so you can decide to stop early.
- If you compute all, you can keep them based on “ratio of variance explained”.

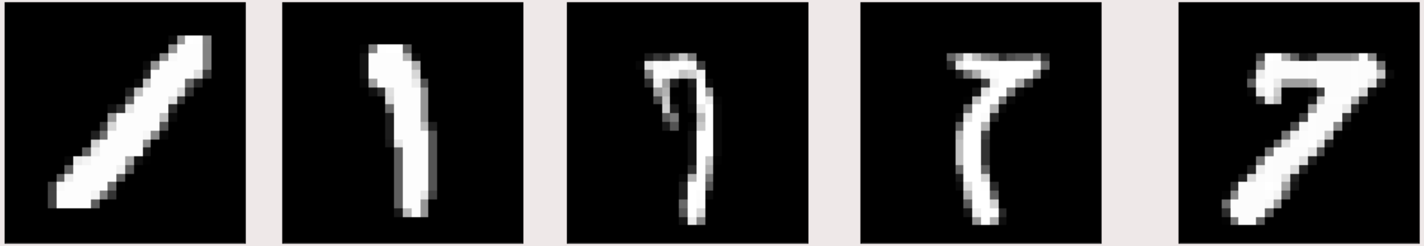




Principal Component Analysis

PCA is most often used to reduce dimensionality in high-dimensional data sets.

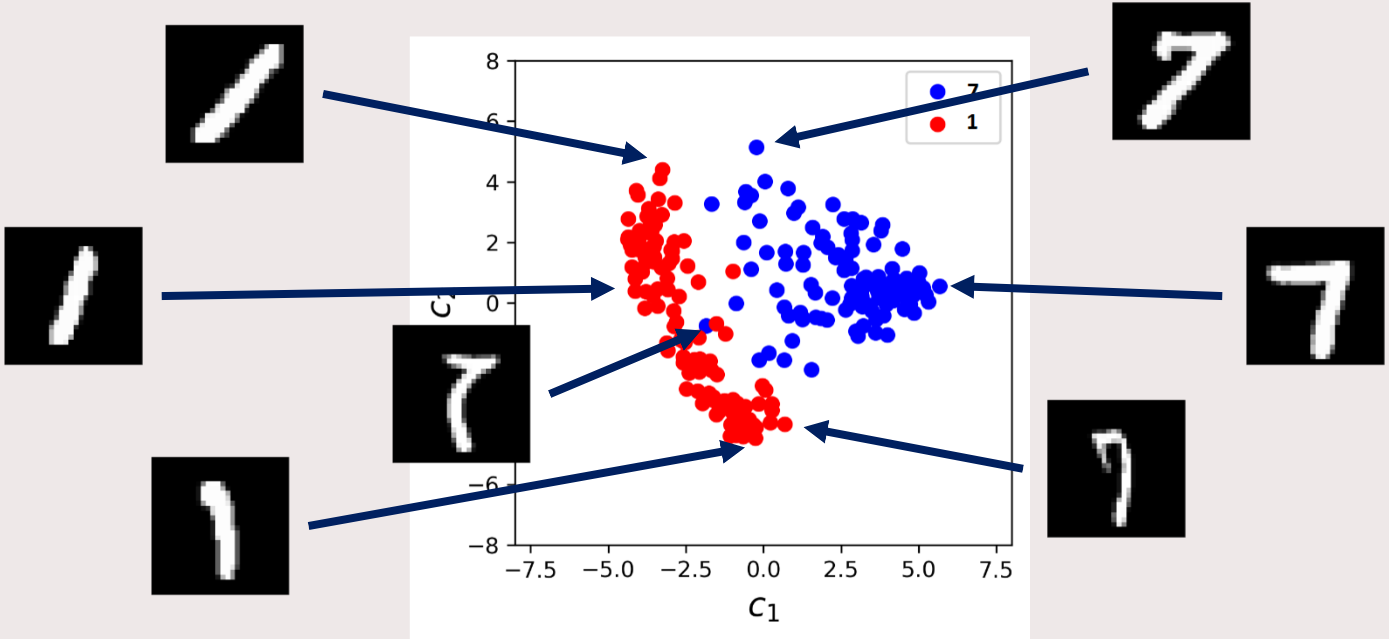
To give you an example, consider a data set of handwritten digits 1 and 7:



If you reshape each 28×28 matrix into a 784×1 vector, you can build a data set of size $784 \times n$ and perform principal component analysis.

Figure are from MNIST Handwritten Digits data set *(link)*.

Principal Component Analysis



Through visual inspection, you get a sense of what directions of variance correspond to.

Principal Component Analysis

Q: What are the “components” in PCA?

1. [✓ The eigenvectors of the covariance matrix.]
2. The eigenvalues of the covariance matrix.
3. The data projected onto the eigenvectors of the covariance matrix.
4. The data projected onto the eigenvectors of the covariance matrix scaled by the eigenvalues of the covariance matrix.

Independent Component Analysis

PCA finds a basis where the projected data becomes uncorrelated, but not independent.

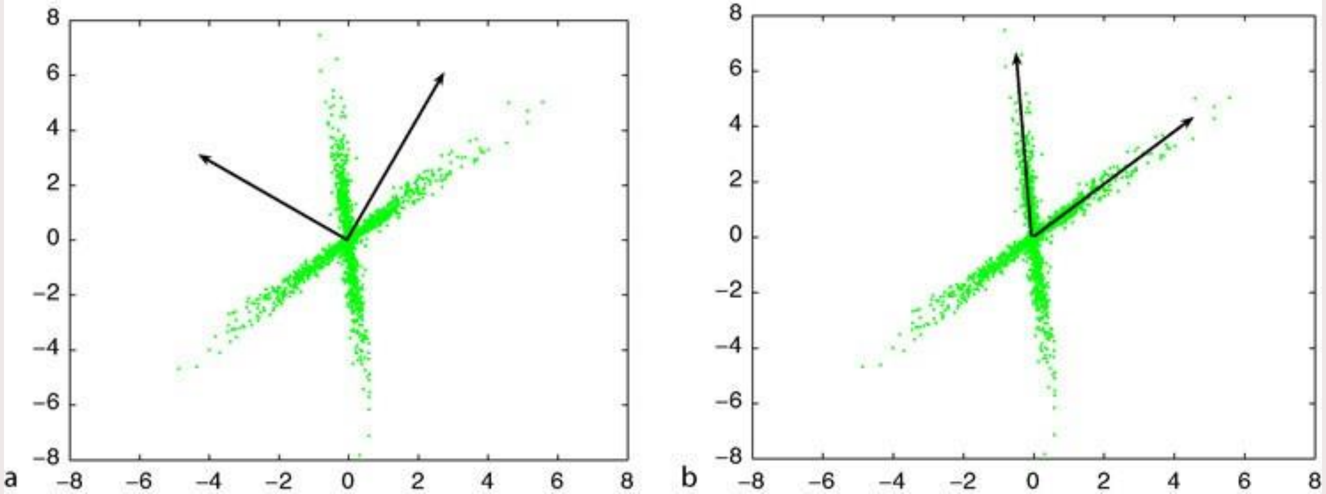


Figure taken from *Choi (2009), Independent Component Analysis (doi).*

Independent Component Analysis

We assume there exists a set S of independent components / sources and that we observe signals that have been mixed linearly by a mixing matrix A:

$$X = AS$$

We presume there exists an unmixing matrix W that generates signals:

$$Y = WX$$

If we find the optimal W, i.e., $W \rightarrow A^{-1}$, then our unmixed data Y would recover the original sources, i.e., $Y \rightarrow S$.

Q: What criterion can we use to find the optimal unmixing matrix W?

Independent Component Analysis

We can use mutual information as it is a measure of how well a set of independent distributions, e.g., $\boldsymbol{p(x)p(y)}$, approximates a joint distribution, $\boldsymbol{p(x,y)}$.

We can rewrite MI as:

$$\begin{aligned}\boldsymbol{I[X,Y]} &= \boldsymbol{H[X] - X[X|Y]} \\ &= \boldsymbol{H[X] + H[Y] - H[X,Y]}\end{aligned}$$

Maximizing the joint entropy $\boldsymbol{H[Y] = H[Y_1, Y_2, \dots]}$ maximizes MI.

- When the joint entropy is maximal, the Y_i's are mutually independent*.

- The joint entropy is maximal when the joint probability density $p(Y_1, Y_2, \dots)$ is uniform*.

**Cover & Thomas (1991), Elements of Information Theory.*

Independent Component Analysis

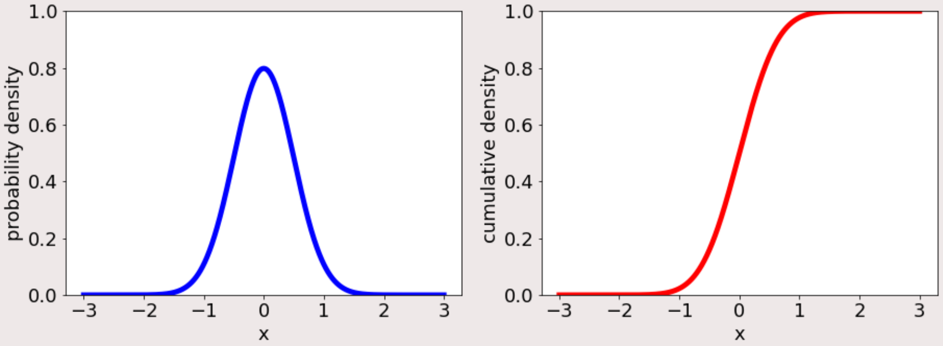
Q: How are we going to make the joint probability density uniform?

There are a few ways to do this, but we will look at *Infomax*.

In Infomax, we optimize the cumulative density instead of the probability density.

Reminder: the probability density function of a random variable is the derivative of its cumulative density function,

$$\boldsymbol{p(x) = \frac{dP(x)}{dx}}.$$



Independent Component Analysis

Let $\mathcal{Y} = g(Y)$ be an invertible nonlinear transformation of the unmixed signals, where g is the cumulative distribution function of Y.

- Here, $\boldsymbol{g}$ is applied to each element of $\boldsymbol{Y}$: $\boldsymbol{g}(\boldsymbol{Y}) = (\boldsymbol{g}(\boldsymbol{Y}_1), \boldsymbol{g}(\boldsymbol{Y}_2), \dots) = (\boldsymbol{\mathcal{Y}}_1, \boldsymbol{\mathcal{Y}}_2, \dots)$.
- Note that independent variables remain independent after invertible transformations.
- We don’t know the actual cumulative distribution function $\boldsymbol{g}$, but we will see later that functions that look sufficiently similar will have the same effect during optimization.

Independent Component Analysis

We want to maximize the joint entropy $H[\mathcal{Y}]$, for which we need an objective function.

The entropy of a single component $\mathcal{Y}_i$ estimated using histograms is:

$$\hat{H}[\mathcal{Y}_i] = -\frac{1}{|\mathcal{Y}_i|} \sum_{y \in \mathcal{Y}_i} \log p_{\mathcal{Y}_i}(y)$$

where $p_{\mathcal{Y}_i}(y)$ refers to the probability of the specific outcome $y \in \mathcal{Y}_i$ and $|\mathcal{Y}_i|$ refers to the number of outcomes of the variable $\mathcal{Y}_i$.

Note that the probability before the logarithm is gone; if we use equally sized bins in the histogram, then this average log-probability will converge to the expected log-probability as the number of observations goes to infinity*.

**Law of large numbers (https://www.statlect.com/asymptotic-theory/law-of-large-numbers).*

Independent Component Analysis

There is a rule to obtain the probability density of a random variable that has been subjected to an invertible nonlinear transformation, e.g., for $u = h(v)$:

$$p(u) = \frac{1}{\left|\frac{dh(v)}{dv}\right|} p(v)$$

where the brackets denote the absolute value.

We can use that rule for our transformed variable:

$$p_{\mathcal{y}_i}(\mathcal{y}) = \frac{1}{\frac{d\mathcal{y}_i}{dY_i}} p_{Y_i}(y)$$

where the absolute value is unnecessary because g is a cdf and cdf's are always positive.

Independent Component Analysis

For the true g, the derivative is $\frac{dy_i}{dY_i} = \frac{dg(Y_i)}{dY_i} = p_{Y_i}(y)$.

We don't know the true g, but we can take a convenient function $\hat{g}$ instead.

A common choice in Infomax is $\hat{g}$ to choose such that:

$$\frac{d\hat{g}(Y_i)}{dY_i} = (1 - \tanh(Y_i))^2$$

Other choices are possible, as long as they resemble a cumulative distribution function.

Independent Component Analysis

Using the same formula for transformation of random variables, we get:

$$p_{Y_i}(y) = \frac{1}{|W|} p_{X_i}(x)$$

where x is an outcome (bin) in the histogram for the i-th data dimension.

This result lets us express:

$$p_{\mathcal{Y}_i}(\mathcal{y}) = \frac{1}{\frac{d\mathcal{Y}_i}{dY_i}} p_{Y_i}(y) = \frac{p_{X_i}(x)}{|W|(1 - \tanh(Wx))^2}$$

Note that the x in the numerator and denominator refer to the same outcome of X_i.

Independent Component Analysis

Plugging this result into the empirical entropy gives:

$$\hat{H}[\mathcal{Y}_i] = -\frac{1}{|X_i|} \sum_{x \in X_i} \log \frac{p_{X_i}(x)}{|W|(1 - \tanh(Wx))^2}$$

where the sum now iterates over outcomes of X_i.

Decomposing the logarithm gives:

$$\hat{H}[\mathcal{Y}_i] = -\frac{1}{|X_i|} \sum_{x \in X_i} \log p_X(x_i) + \frac{1}{|X_i|} \sum_{x \in X_i} \log(1 - \tanh(Wx))^2 + \log |W|$$

which is a function we can evaluate.

Independent Component Analysis

The first term, $-\frac{1}{|X_i|}\sum_{x \in X_i} \log p_X(x_i)$, does not depend on W and can be dropped.

Considering *all* K components gives our objective function:

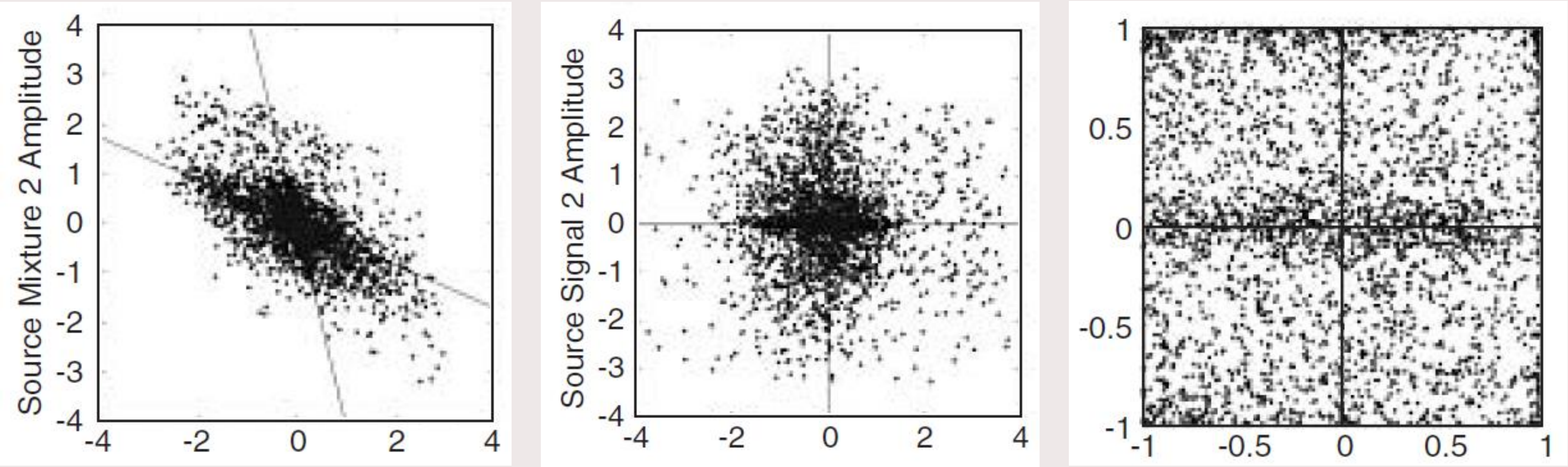
$$C(W) = \sum_{i=1}^{K} \frac{1}{|X_i|} \sum_{x \in X_i} \log(1 - \tanh(Wx))^2 + \log|W|$$

We can maximize this objective to obtain our optimal unmixing matrix:

$$\widehat{W} = \arg\max_{W} \; C(W)$$

This $\widehat{W}$ is not necessarily identical to the true $W = A^{-1}$ (e.g., it may be permuted) but *will* produce independent components.

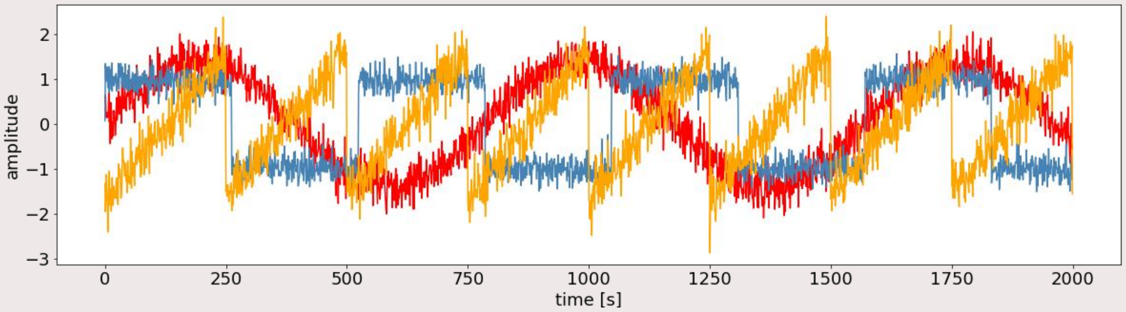
Summary Infomax ICA

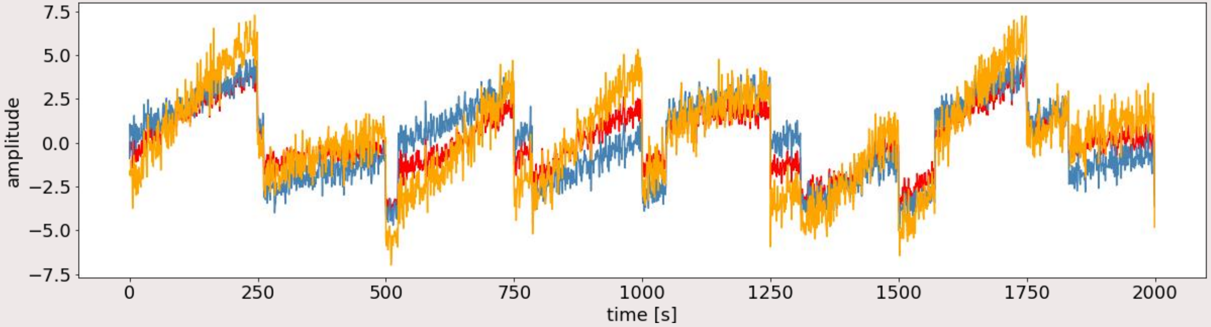


$$\boldsymbol{X} \longrightarrow \widehat{\boldsymbol{W}}\boldsymbol{X} \longrightarrow \widehat{\boldsymbol{g}}(\widehat{\boldsymbol{W}}\boldsymbol{X})$$

Figure taken from *Stone (2004), Independent Component Analysis: a tutorial introduction.*

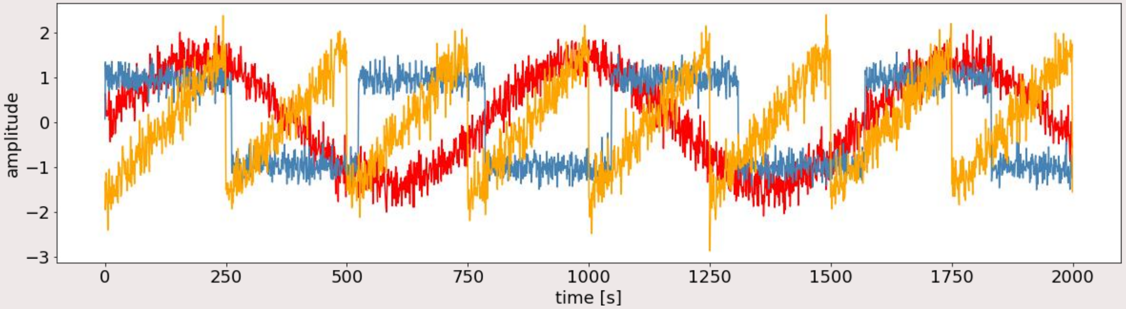
Blind source separation

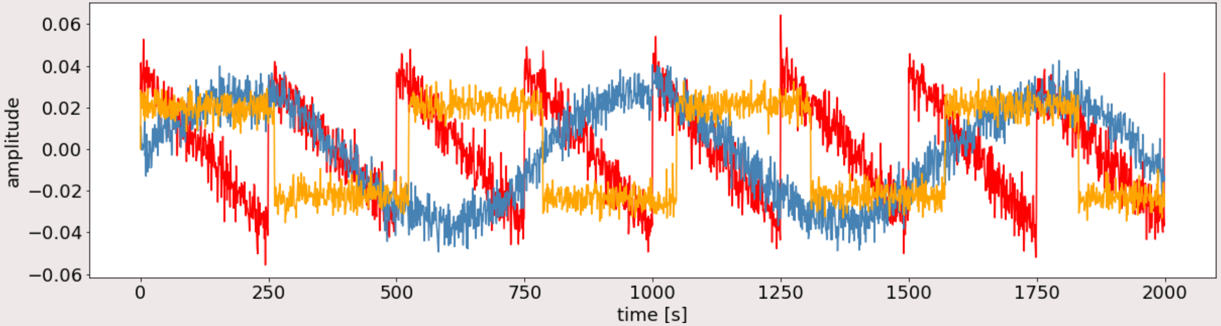




Data generated according scikit-learn’s *Blind source separation using FastICA* (*link*).

Blind source separation





Data generated according scikit-learn’s *Blind source separation using FastICA* (*link*).

Summary: feature extraction

New features may be constructed from existing features using designed transformations.

One could use simple characteristic functions (e.g., *min, max, mean*) whenever there is context- / task- / setting-dependent information available.

We could find an orthogonal basis pointing in the directions of maximal variance.

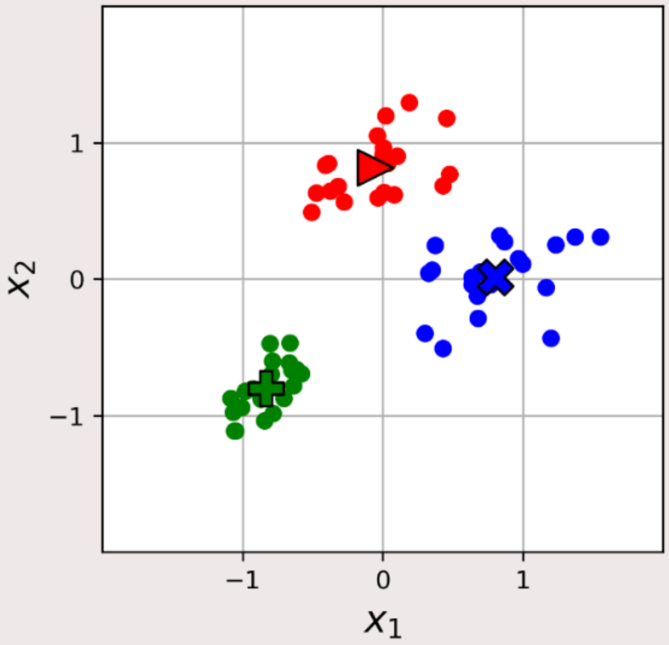
- This is excellent for data compression (i.e., gives the least amount of variance lost).

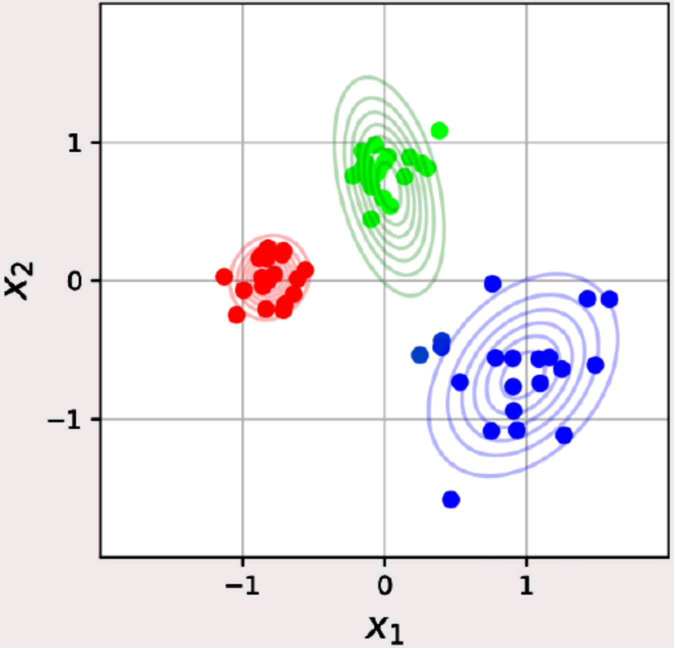
We can find a linear basis where the data becomes as independent as possible, by maximizing the entropy of the unmixed data.

- This is useful when we suspect that the data was generated by independent sources (e.g., speakers in a room, EEG, seismology, music).

Questions

Next time: clustering and unsupervised learning


1
0
−1
−1
0
1
x_2
x_1


1
0
−1
−1
0
1
x_2
x_1