# (Concepts of) Machine Learning

## Lecture 3: Features engineering and learning paradigms

George Magoulas
gmagoulas@dcs.bbk.ac.uk

# Outline

- Selecting features for machine learning
  - Feature selection based on statistical testing
  - Class separability measures
  - Features subset selection
- Feature generation/selection through learning
- Summary

# Selecting features for ML

- Large $l$ has a three-fold disadvantage:
  - High computational demands
  - Low generalization performance
  - Poor error estimates

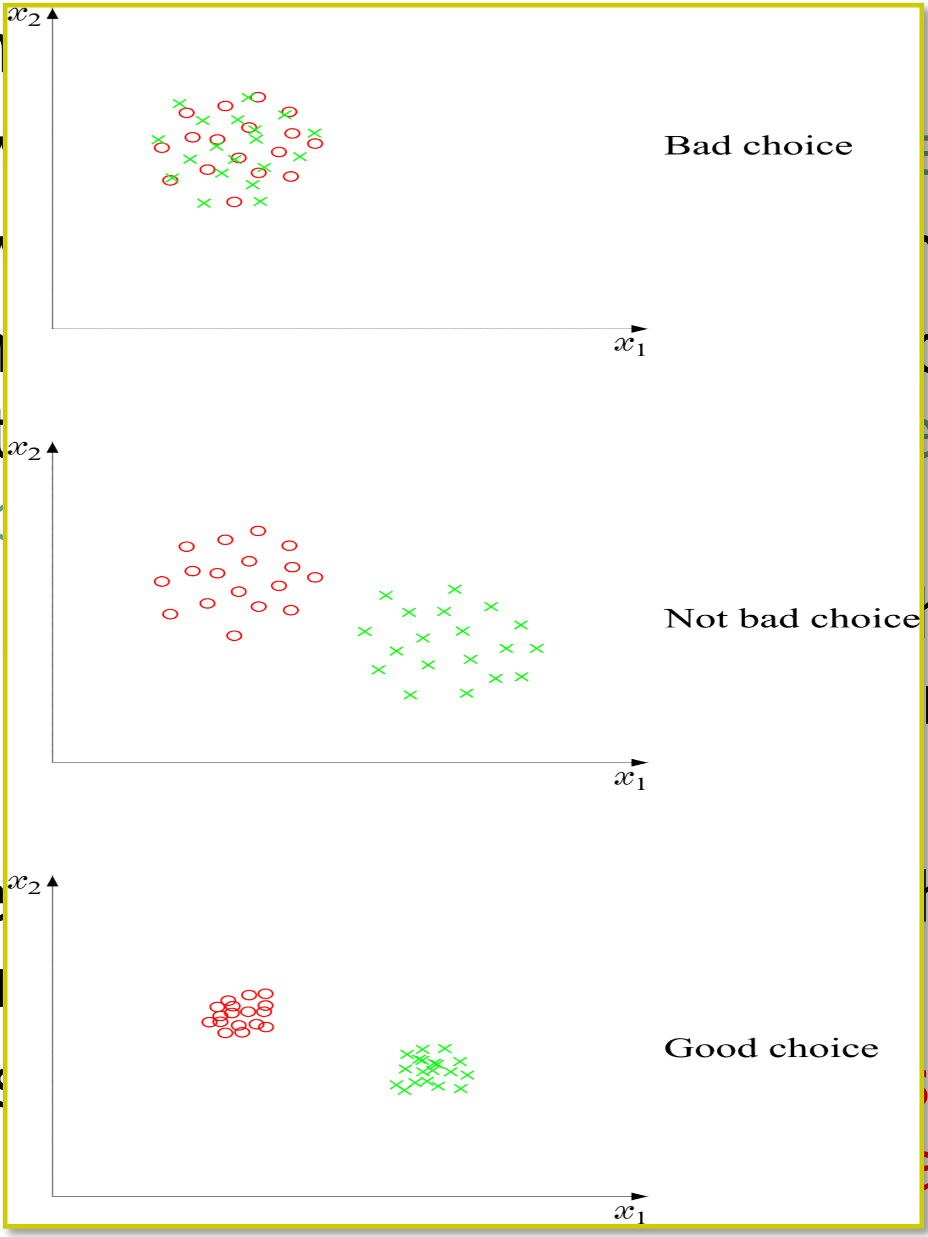
*General case:*

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_d \end{bmatrix}$$

is in the *d*-dimensional domain of the feature vectors

- Given $N$ training patterns
  - $l$ must be large enough to learn
    - what makes classes different (e.g. apples/bananas)
    - what makes patterns in the same class similar
  - $l$ must be small enough not to learn what makes patterns of the same class different (e.g. red/green apple)
  - In practice, $l < N/a$, $a$ in $[2, 10]$ has been reported to be a sensible choice for a number of cases
- Once $l$ has been decided, choose the $l$ most informative features
  - Best: **Large** between class distance,
    **Small** within class variance

- Given $N$ training patterns
  - $l$ m
    - w e.g. apples/bananas)
    - w me class similar
  - $l$ n o learn what makes pat erent (e.g. red/green app
  - In has been reported to be r of cases
- Once he $l$ most informative featu
  - Bes s distance, ariance



# Feature selection (apply after preprocessing)

- Discard individual features with poor information content, i.e. select most promising features

- The remaining information rich features are examined jointly as vectors, i.e. test feature combinations discrimination ability

# Feature Selection based on statistical hypothesis testing

- The Goal: **For each individual feature**, find whether the values, which the feature takes for the different classes, differ significantly.
  That is, answer
  - $H_1 : \ \theta_1 \neq \theta_0$: The values differ significantly
  - $H_0 : \ \theta_1 = \theta_0$: The values do not differ significantly

  If they do not differ significantly reject feature from subsequent stages.
- Hypothesis Testing Basics- http://en.wikipedia.org/wiki/Statistical_hypothesis_testing

# example

A feature $x$ is measured $N$ times and we calculate

$$\bar{x} = 1.35$$

Test the hypothesis

$$H_0 : \mu = \hat{\mu} = 1.4$$

$$H_1 : \mu \neq \hat{\mu}$$

*Why is that useful?*

If the values that the feature takes do not differ significantly from the mean, one may decide not to measure this feature in subsequent data processing stages.

# example

A feature $x$ is measured $N$ times and we calculate

the mean value it takes in each class

Test the hypothesis

$$H_0 : \Delta\mu = \mu_1 - \mu_2 = 0$$

$$H_1 : \Delta\mu \neq 0$$

Why is that useful?

If the zero hypothesis is rejected, this feature is important

- The steps:
  - $N$ measurements are known $x_i, i = 1,2,\dots, N$
  - Define a function of them

    $$q = f(x_1, x_2, \dots, x_N):$$ test statistic

    http://en.wikipedia.org/wiki/Test_statistic

    so that $\boxed{p_q(q;\theta)}$ is easily parameterised in terms of $\theta$.
  - Let $D$ be an interval, where $q$ has a high probability to lie under $H_0$, i.e., $p_q(q|\theta_0)$
  - Let $\overline{D}$ be the complement of $D$

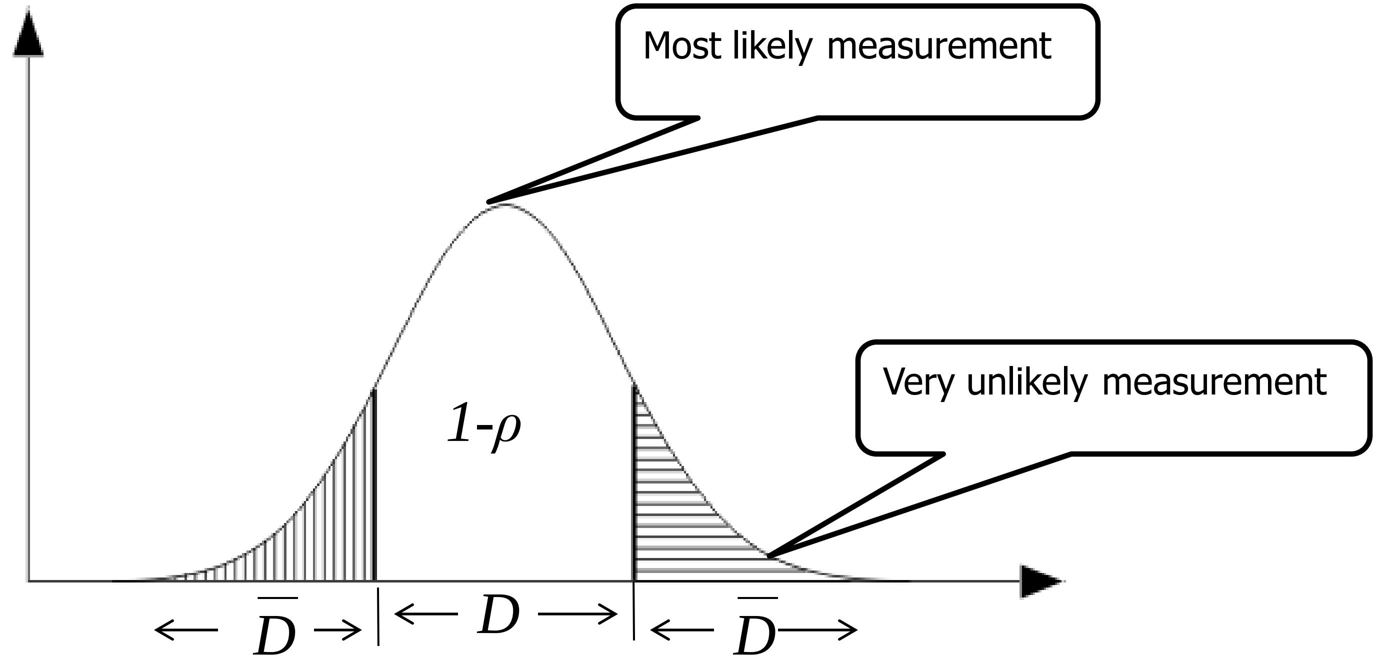
    $D \longrightarrow$ Acceptance Interval

    $\overline{D} \longrightarrow$ Critical Interval
  - If $q$, resulting from $x_1, x_2, \dots, x_N$, lies in $D$ we accept $H_0$, otherwise we reject it.

- Probability of an error

$$p_q(q \in \overline{D} | H_0) = \rho$$

the probability density of each outcome, computed under the null hypothesis.


$p(q|H_0)$
$1\text{-}\rho$
$\leftarrow \overline{D} \rightarrow$
$\leftarrow D \longrightarrow$
$\leftarrow \overline{D} \longrightarrow$

Most likely measurement

Very unlikely measurement

- In practice, $\rho$ is preselected and it is known as the significance level.

# Application to features: The known variance case

- Let $x$ be a random variable and the experimental samples, $x_i = 1,2,\ldots,N$ , are assumed mutually independent. Also let

$$E[x] = \mu$$

$$E[(x-\mu)^2] = \sigma^2$$

- Compute the sample mean

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

- This is also a random variable with mean value

$$E[\bar{x}] = \frac{1}{N}\sum_{i=1}^{N} E[x_i] = \mu$$

That is, it is an Unbiased Estimator of the mean of $x$

- The variance $\sigma^2_{\bar{x}}$

$$E[(\bar{x}-\mu)^2] = E[(\frac{1}{N}\sum_{i=1}^{N} x_i - \mu)^2]$$

$$= \frac{1}{N^2}\sum_{i=1}^{N} E[(x_i-\mu)^2] + \frac{1}{N^2}\sum_{i}\sum_{j} E[(x_i-\mu)(x_j-\mu)]$$

Due to independence of the samples

$\sigma^2_{\bar{x}} = \frac{1}{N}\sigma^2_x$ i.e. largest the no of measurements, the smaller the variance around the true mean

That is, it is Asymptotically Efficient

- Hypothesis test

$$H_1 : E[x] \neq \hat{\mu}$$

$$H_0 : E[x] = \hat{\mu}$$

- Test Statistic: Define the variable

$$\boxed{q = \frac{\bar{x}-\hat{\mu}}{\sigma/\sqrt{N}}}$$

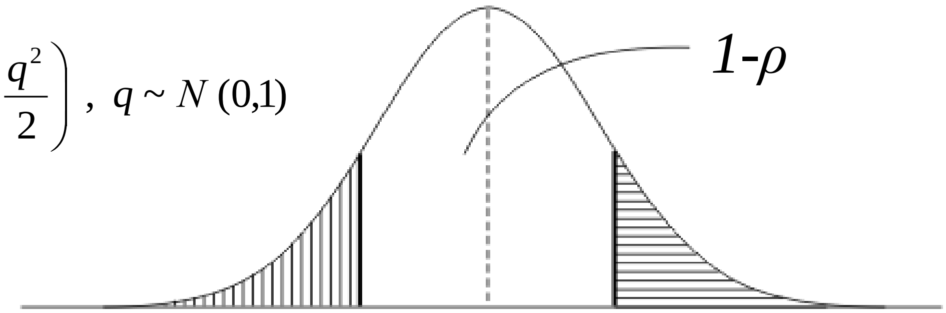
(as $N$ gets larger, the distribution of the difference between the sample average and its limit $\mu$, when multiplied by the factor $\sqrt{N}$, approximates the normal distribution with mean 0 and variance $\sigma^2$. )

under $H_0$ the probability density function is approximated by a Gaussian

$$p_{\bar{x}}(\bar{x}) = \frac{\sqrt{N}}{\sqrt{2\pi\sigma}} \exp\left( -\frac{N(\bar{x} - \hat{\mu})^2}{2\sigma^2} \right), \quad N(\hat{\mu}, \frac{\sigma^2}{N})$$

- The decision steps
  - Compute $q$ from $x_i$, $i=1,2,\ldots,N$
  - Choose significance level $\rho$
  - Compute from $N(0,1)$ tables $D=[-x_\rho, x_\rho]$

$$p_q(q)=\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{q^2}{2}\right),\ q\sim N(0,1)$$



  - $$\begin{array}{ll} \text{if} \quad q\in D & \text{accept}\ H_0 \\ \text{if} \quad q\in \overline{D} & \text{reject}\ H_0 \end{array}$$

- An example: A random variable $x$ has variance $\sigma^2=(0.23)^2$. $N=16$ measurements are obtained giving $x=1.35$. The significance level is $\rho=0.05$.

Test the hypothesis

$$H_0:\mu=\hat{\mu}=1.4$$

$$H_1:\mu\neq\hat{\mu}$$

- Since σ² is known, $q = \dfrac{\bar{x} - \hat{\mu}}{\sigma / 4}$ is $N\,(0,1)$.

  From tables, we obtain the values with acceptance intervals $[-x_\rho, x_\rho]$ for normal $N\,(0,1)$

| ***1-ρ*** | 0.8 | 0.85 | 0.9 | 0.95 | 0.98 | 0.99 | 0.998 | 0.999 |
|---|---|---|---|---|---|---|---|---|
| $\boldsymbol{x_\rho}$ | 1.28 | 1.44 | 1.64 | 1.96 | 2.32 | 2.57 | 3.09 | 3.29 |

- Thus

$$\text{Prob}\left\{-1.967 < \frac{\bar{x} - \hat{\mu}}{0.23/4} < 1.967\right\} = 0.95$$

or

$$\text{Prob}\left\{-0.113 < \bar{x} - \hat{\mu} < 0.113\right\} = 0.95$$

or

$$\boxed{\text{Prob}\left\{1.237 < \hat{\mu} < 1.463\right\} = 0.95}$$

- Since $\hat{\mu} = 1.4$ *lies* within the above ***acceptance interval***, we accept $H_0$, i.e.,

$$\mu = \hat{\mu} = 1.4$$

The interval [1.237, 1.463] is also known as confidence interval at the $1\text{-}\rho=0.95$ level.

We say that: there is no evidence at the 5% level that the mean value is not equal to $\hat{\mu}$

*Thus, the values, which the feature takes do not differ significantly from the mean. If they do not differ significantly one may decide not to measure this feature in subsequent data processing stages.*

# The Unknown Variance Case

- Estimate the variance. The estimate

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2$$

is unbiased (independence of data samples), i.e.

$$E[\hat{\sigma}^2] = \sigma^2$$

- Define the test statistic

$$q = \frac{\overline{x} - \mu}{\hat{\sigma}/\sqrt{N}}$$

- This $q$ is no longer $N(0,1)$ distribution. If $x$ is Gaussian, then

  $q$ follows a t-distribution, with $N$-1 degrees of freedom

- An example:

  $x$ is Gaussian, $N = 16$, obtained from measurements,

  $\bar{x} = 1.35$ and $\hat{\sigma}^2 = (0.23)^2$. Test the hypothesis

  $H_0 : \ \mu = \hat{\mu} = 1.4$

  at the significance level $\rho = 0.025$.

# Table of acceptance intervals for $t$-distribution

| Degrees of Freedom | 1-ρ | 0.9 | 0.95 | 0.975 | 0.99 |
|---|---|---|---|---|---|
| 12 | | 1.78 | 2.18 | 2.56 | 3.05 |
| 13 | | 1.77 | 2.16 | 2.53 | 3.01 |
| 14 | | 1.76 | 2.15 | 2.51 | 2.98 |
| 15 | | 1.75 | 2.13 | 2.49 | 2.95 |
| 16 | | 1.75 | 2.12 | 2.47 | 2.92 |
| 17 | | 1.74 | 2.11 | 2.46 | 2.90 |
| 18 | | 1.73 | 2.10 | 2.44 | 2.88 |

$$\text{Prob}\left\{-2.49 < \frac{\bar{x} - \hat{\mu}}{\hat{\sigma}/4} < 2.49\right\}$$

$$1.207 < \hat{\mu} < 1.493$$

$$\text{Thus, } \hat{\mu} = 1.4 \text{ is accepted}$$

# Application of *t*-test in Feature Selection

- The goal here is to test against zero the difference $\mu_1$-$\mu_2$ of the respective means in classes $\omega_1$, $\omega_2$ of a single feature. Assume statistical independence

- Let $x_i$ $i=1,\ldots,N$ , the values of a feature in $\omega_1$

- Let $y_i$ $i=1,\ldots,N$ , the values of the same feature in $\omega_2$

- Assume in both classes $\sigma_1^2=\sigma_2^2=\sigma^2$ (unknown or not)

- The test becomes

$$H_0: \Delta\mu=\mu_1-\mu_2=0$$

$$H_1: \Delta\mu\neq 0$$

- Define

  $z=x-y$

- Obviously

  $E[z]=\mu_1-\mu_2$

- Define the average

  $$\bar{z}=\frac{1}{N}\sum_{i=1}^{N}(x_i-y_i)=\bar{x}-\bar{y}$$

- Known Variance Case: Define

  $$q=\frac{(\bar{x}-\bar{y})-(\hat{\mu}_1-\hat{\mu}_2)}{\sigma\sqrt{\frac{2}{N}}}$$

- This is $N(0,1)$ and one follows the procedure as before.

- Unknown Variance Case:
  Define the test statistic

$$q = \frac{(\bar{x} - \bar{y}) - (\mu_1 - \mu_2)}{S_z \sqrt{\frac{2}{N}}}$$

$$S_z^2 = \frac{1}{2N-2} \left( \sum_{i=1}^{N} (x_i - \bar{x})^2 + \sum_{i=1}^{N} (y_i - \bar{y})^2 \right)$$

  - $q$ is t-distribution with $2N$-2 degrees of freedom,
  - Then apply appropriate tables as before.

- Example: The values of a feature in two classes are:

  $\omega_1$: 3.5, 3.7, 3.9, 4.1, 3.4, 3.5, 4.1, 3.8, 3.6, 3.7

  $\omega_2$: 3.2, 3.6, 3.1, 3.4, 3.0, 3.4, 2.8, 3.1, 3.3, 3.6

  Test if the mean values in the two classes differ significantly, at the significance level $\rho$=0.05

- We have

$$\omega_1: \quad \bar{x} = 3.73, \; \hat{\sigma}_1^2 = 0.0601$$

$$\omega_2: \quad \bar{y} = 3.25, \; \hat{\sigma}_2^2 = 0.0672$$

For $N$=10

$$S_z^2 = \frac{1}{2}(\hat{\sigma}_1^2 + \hat{\sigma}_2^2)$$

$$q = \frac{(\bar{x} - \bar{y}) - 0}{S_z \sqrt{\frac{2}{10}}}$$

$$\boxed{q = 4.25}$$

- From the table of the t-distribution with $2N$-$2$=18 degrees of freedom and $\rho$=0.05, we obtain $D$=[-2.10,2.10] and since $q$=4.25 is outside $D$, $H_1$ is accepted and *the feature is selected*.

- We have

$$\omega_1: \quad \bar{x} = 3.73, \ \hat{\sigma}_1^2 = 0.0601$$

$$\omega_2: \quad \bar{y} = 3.25, \ \hat{\sigma}_2^2 = 0.0672$$

For $N$=10

$$S_z^2 = \frac{1}{\;}$$

$$q$$

$$\boxed{q = 4.25}$$

in practice the variances may not be the same in the two classes. This becomes the object of another hypothesis test that compares variance instead of mean- the so-called F-distribution and the related tables
should be used

- From the table of the t-distribution with $2N$-$2$=18 degrees of freedom and $\rho$=0.05, we obtain $D$=[-2.10,2.10] and since $q$=4.25 is outside $D$, $H_1$ is accepted and *the feature is selected*.

# Class separability measures

So far we looked at individual features. What happens if there are existing correlations among the features?

- **Two features may be rich in information, but if they are highly correlated we need not consider them both**. To this end, in order to search for possible correlations, we consider features jointly as elements of vectors. This can be used to:
  - Produce the “best” vector of $l$ features to be used. This is dictated by the specific problem (e.g., the number, $N$, of available training patterns and the type of the classifier to be adopted).
  - Transform the original data on the basis of an optimality criterion in order to come up with features offering high classification power.

- One can:
  - Use different feature combinations to form the feature vector. Train the classifier, and choose the combination resulting in the best classifier performance.

    A disadvantage of this approach is the high complexity. Also, local minima, may give misleading results.
- Next, we adopt a class separability measure and choose the best feature combination against this cost- that is independent of the classifier.

Let $\mathbf{x}$ be the current feature combination vector.

**(i) Divergence cost.** To see the rationale behind this cost, consider the two – class case.

- Obviously, if on the average the value of $\ln \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)}$ is close to zero, then $\mathbf{x}$ should be a

  poor feature combination (overlapped classes). Define mean value over each class:

- $$D_{12} = \int_{-\infty}^{+\infty} p(\mathbf{x}|\omega_1) \ln \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} dx$$
- $$D_{21} = \int_{-\infty}^{+\infty} p(\mathbf{x}|\omega_2) \ln \frac{p(\mathbf{x}|\omega_2)}{p(\mathbf{x}|\omega_1)} dx$$
- $$d_{12} = D_{12} + D_{21}$$

  $d_{12}$ is known as the divergence and can be used as a class separability measure.

- For the multi-class case, define $d_{ij}$ for every pair of classes $\omega_i$, $\omega_j$ and the average divergence is defined as

$$d = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\omega_i) P(\omega_j) d_{ij}$$

$$d_{ij}(x_1, x_2, \cdots x_l) = \sum_{r=1}^{l} d_{ij}(x_r)$$

https://en.wikipedia.org/wiki/Mahalanobis_distance

- Some properties:

$$d_{ij} \geq 0$$

$$d_{ij} = 0, \text{if } i = j$$

$$d_{ij} = d_{ji}$$

- Large values of $d$ are indicative of good feature combination (it means that the particular combination allows to separate classes accurately).

# (ii) Scatter Matrices

These are used as a measure of the way data are scattered in the respective feature space.

- **Within-class** scatter matrix: $S_W = \sum_{i=1}^{M} P_i S_i$

where $S_i = E\left[(\boldsymbol{x}-\boldsymbol{\mu}_i)(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\right] \approx \dfrac{1}{n_i}\sum_{\boldsymbol{x}\in\omega_i}(\boldsymbol{x}-\boldsymbol{\mu}_i)(\boldsymbol{x}-\boldsymbol{\mu}_i)^T$

https://en.wikipedia.org/wiki/Covariance_matrix

is the ***scatter matrix of class*** $\omega_i$ , and the *a priori* probability of class $\omega_i$ is:

$$P_i \equiv P(\omega_i) \approx \frac{n_i}{N}$$

Total number of sample points

$n_i$ the number of training samples in $\omega_i$.

$\text{trace}\{S_W\}$ is a measure of the **average variance** of the features.

- **Between-class** scatter matrix

$$S_B = \sum_{i=1}^{M} P_i \left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)\left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)^T$$

where the global mean vector is: $\boldsymbol{\mu}_0 = \sum_{i=1}^{M} P_i \boldsymbol{\mu}_i$

$\mathrm{trace}\{S_B\}$ is a measure of the **average distance** of the mean of each class from the respective global value.

- **Mixture scatter** matrix

$$S_M = E\left[\left(\boldsymbol{x} - \boldsymbol{\mu}_0\right)\left(\boldsymbol{x} - \boldsymbol{\mu}_0\right)^{\mathrm{T}}\right] \approx \frac{1}{N}\sum_{\boldsymbol{x}} \left(\boldsymbol{x} - \boldsymbol{\mu}_0\right)\left(\boldsymbol{x} - \boldsymbol{\mu}_0\right)^{\mathrm{T}}$$

$$= \frac{1}{N}\sum_{i=1}^{M}\sum_{\boldsymbol{x}\in\omega_i} \left(\boldsymbol{x} - \boldsymbol{\mu}_i\right)\left(\boldsymbol{x} - \boldsymbol{\mu}_i\right)^T + \frac{1}{N}\sum_{i=1}^{M}\sum_{\boldsymbol{x}\in\omega_i} \left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)\left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)^{\mathrm{T}}$$

$$= \sum_{i=1}^{M} \frac{n_i}{N} S_i + \sum_{i=1}^{M} \frac{n_i}{N} \left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)\left(\boldsymbol{\mu}_i - \boldsymbol{\mu}_0\right)^{\mathrm{T}}$$

which means that: $S_M = S_W + S_B$

# Measures based on Scatter Matrices

- $$J_1 = \frac{\operatorname{trace}\{S_M\}}{\operatorname{trace}\{S_W\}}$$
- $$J_2 = \frac{|S_M|}{|S_W|} = \left|S_W{}^{-1} S_M\right|$$

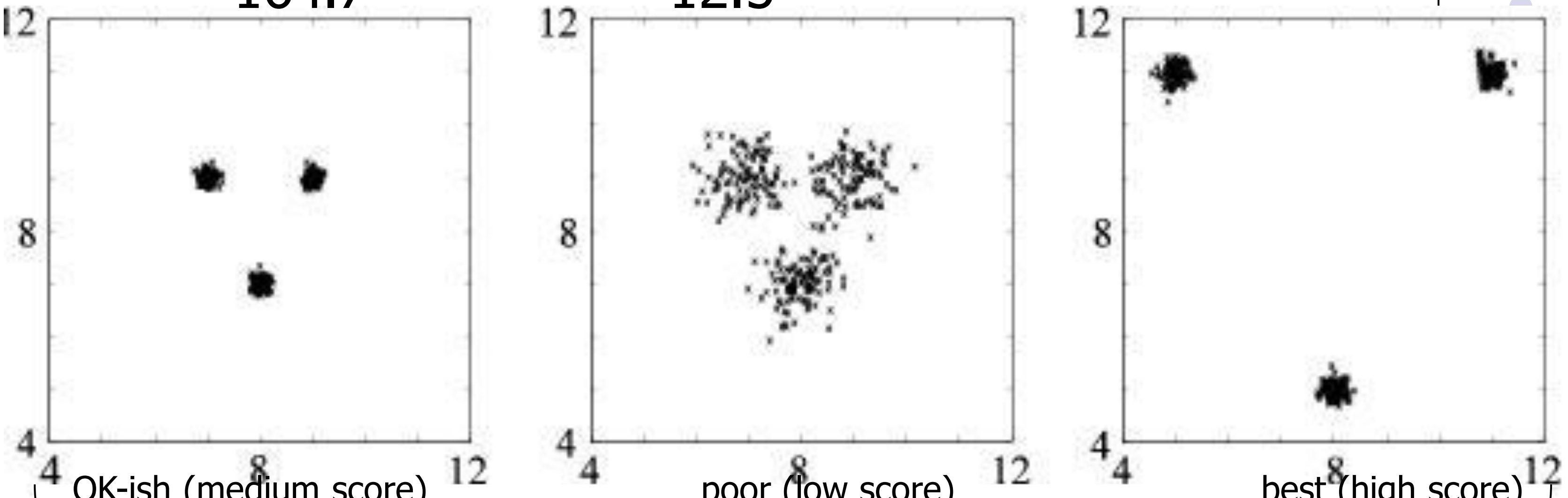
  determinant
- $$J_3 = \operatorname{trace}\left\{S_W{}^{-1} S_M\right\}$$

**These are measures of separability among all classes and can be used as criteria in feature selection**, i.e. to obtain $l$ from the $L$ features to form a sub-feature space in which the separability is maximised.

The trace is equal to the sum of the eigenvalues, while the determinant is equal to their product.

- $\operatorname{trace}\{S_M\}$ is the sum of variances of the features around their respective global mean.
- $\operatorname{trace}\{S_W\}$ is a measure of the average, over all classes, variance of the features.
- Other criteria are also possible, by using various combinations of $S_M$, $S_B$, $S_W$, as suggested in the literature.

164.7 12.5 620.9



OK-ish (medium score) poor (low score) best (high score)

$J_3$ example

The $J_1$, $J_2$, $J_3$ criteria take high values for cases where:

- Data are grouped together within each class (around their mean).
- The means of the various classes are far from each other .

# Feature subset selection: how to combine features

Trying to form all possible combinations of $\lambda$ features from an original set of $m$ selected features is a computationally hard task. Thus, a number of suboptimal searching techniques have been derived.

**(i) Sequential backward selection**. Let $x_1,\ x_2,\ x_3,\ x_4$ be the available features ($m$=4). The procedure consists of the following steps:

- Adopt a class separability criterion (could also be the error rate of the respective classifier). Compute its value for ALL features considered jointly $[x_1, x_2, x_3, x_4]^T$.

- Eliminate one feature and for each of the possible resulting combinations, that is $[x_1,\ x_2,\ x_3]^{\mathrm{T}}$, $[x_1,\ x_2,\ x_4]^{\mathrm{T}}$, $[x_1,\ x_3,\ x_4]^T$, $[x_2,\ x_3,\ x_4]^T$, compute the class separability criterion value $C$. Select the best combination, say $[x_1, x_2, x_3]^{\mathrm{T}}$.

- From the above selected feature vector eliminate one feature and for each of the resulting combinations, $[x_1, x_2]^T$, ..$[x_2, x_3]^T$, $[x_1, x_3]^T$ compute $C$ and select the best combination.

The above selection procedure shows how one can start from $m$ features and end up with the "best" $\lambda$ ones. Obviously, the choice is suboptimal. The number of required calculations is:

$$1+\frac{1}{2}\big((m+1)m-\lambda(\lambda+1)\big)$$

In contrast, a full search requires: $\binom{m}{\lambda}=\frac{m!}{\lambda!(m-\lambda)!}$

(eg. $m$=20,$\lambda$=5; 15504 combinations)

**(ii) Sequential forward selection**. the reverse procedure is followed.

- Compute $C$ for each feature.

Select the “best” one, say $x_1$

- For all possible 2-D combinations of $x_1$, i.e., $[x_1, x_2]$, $[x_1, x_3]$, $[x_1, x_4]$ compute $C$ and choose the best, say $[x_1, x_3]$.

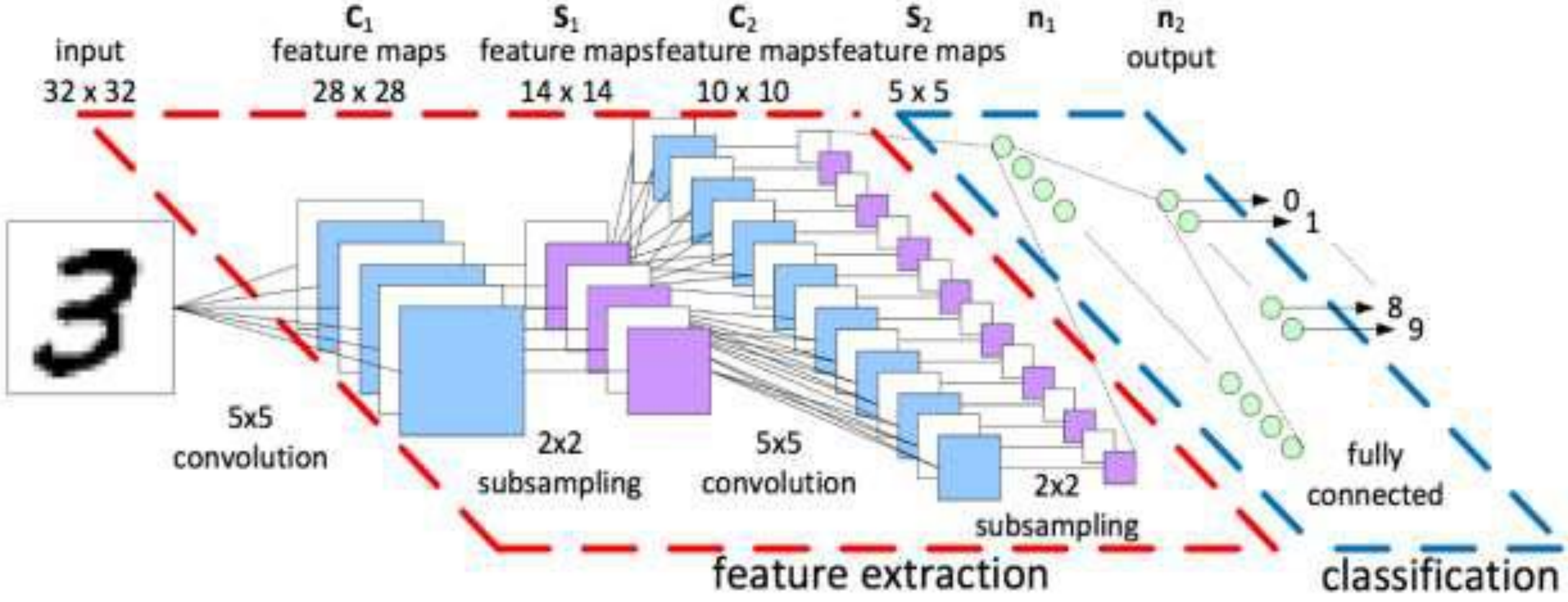
- For all possible 3-D combinations of $[x_1,x_3]$, e.g., $[x_1, x_3, x_2]$, $[x_1, x_{3,}\ x_4]$, etc., compute $C$ and choose the best one.

The above procedure is repeated till the “best” vector with $\lambda$ features has been formed. This is also a suboptimal technique, requiring:

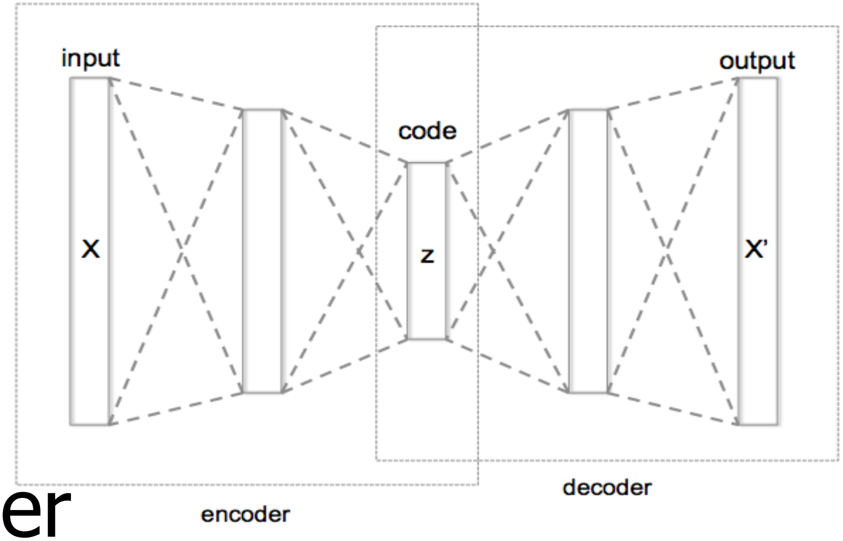
$$\lambda m - \frac{\lambda(\lambda-1)}{2}$$

operations.

# Feature generation/selection through learning



# Autoencoder



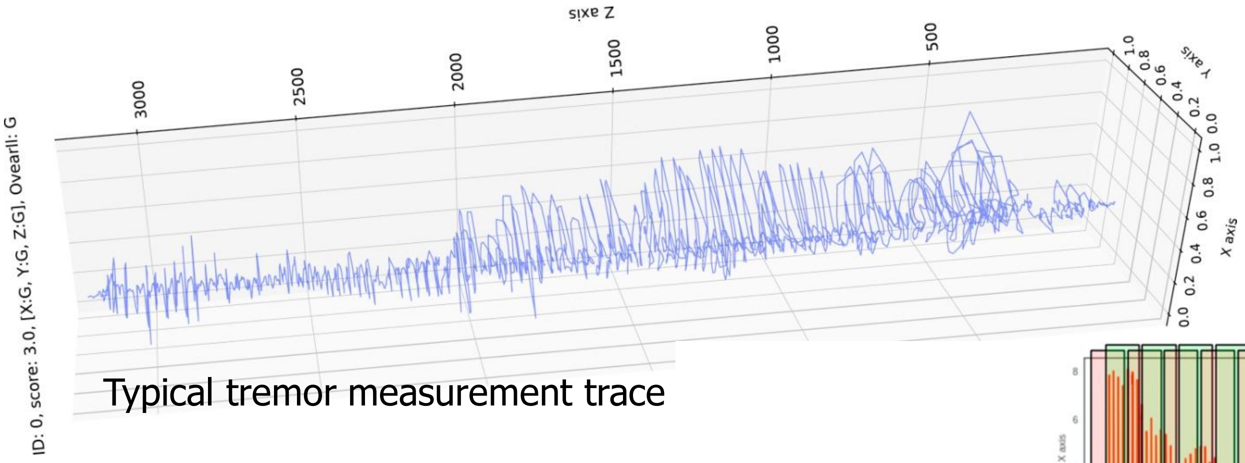
When one linear hidden layer

is used, then is similar to PCA. Upon convergence, the weight vectors of the *h* neurons in the hidden layer form a basis for the space spanned by the first *h* principal components. Unlike PCA, it will not necessarily produce orthogonal vectors (principal components can calculated via singular value decomposition)

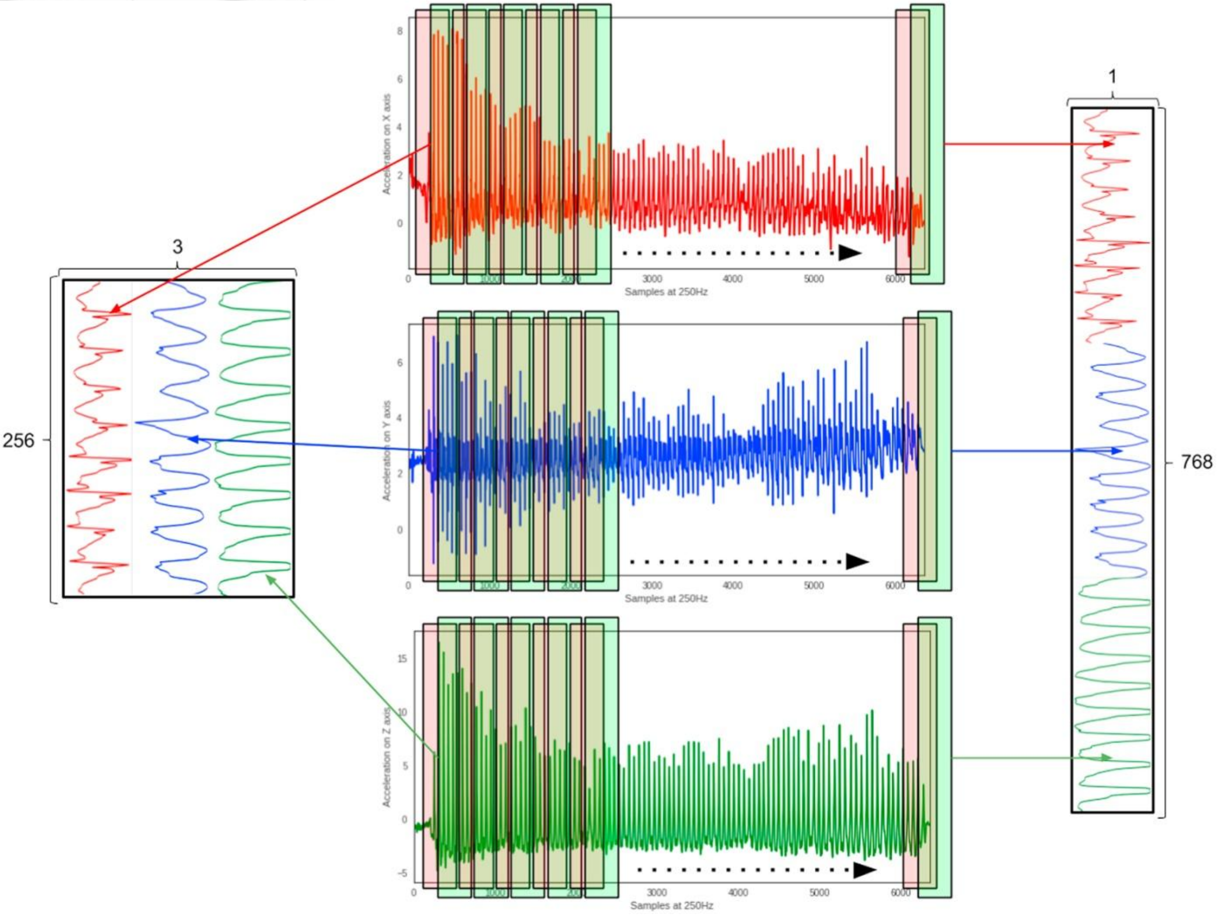
# Depth and abstraction

- (1) deep architectures promote the re-use of features, and
- (2) deep architectures can potentially lead to progressively more abstract features at higher layers of representations

# Parkinson’s tremor measurements

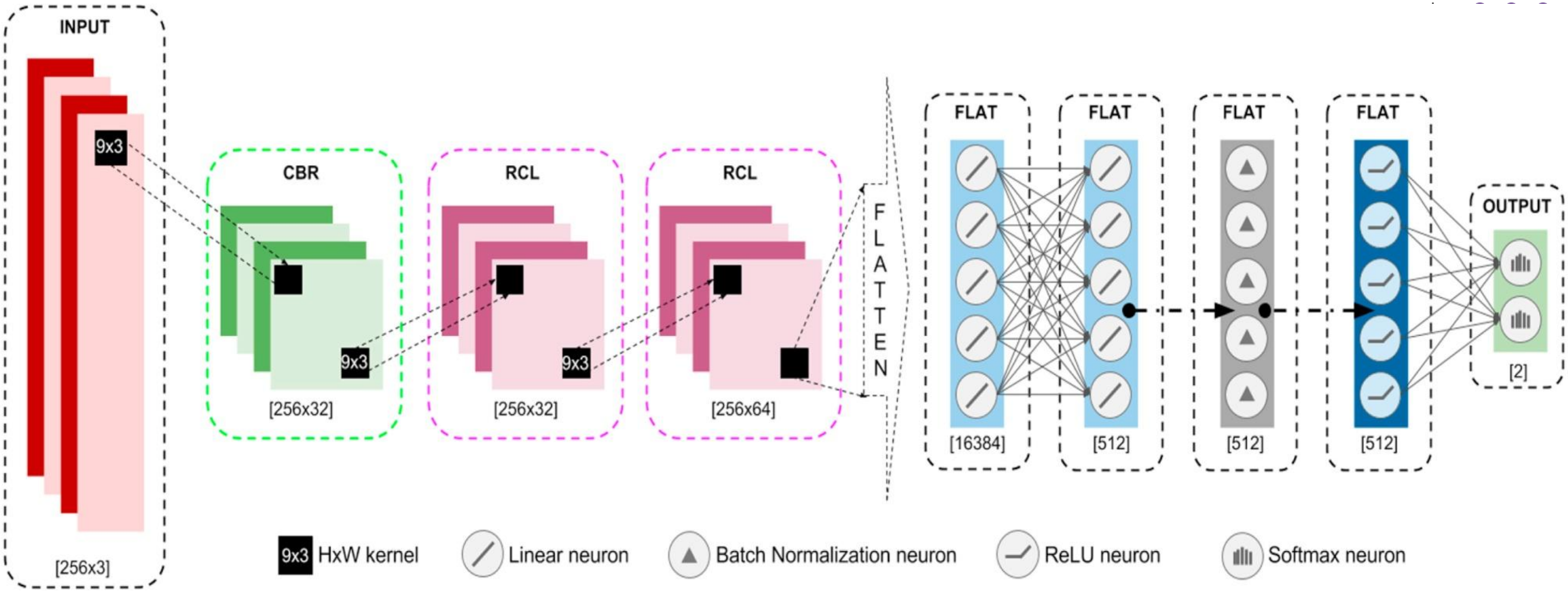


Typical tremor measurement trace

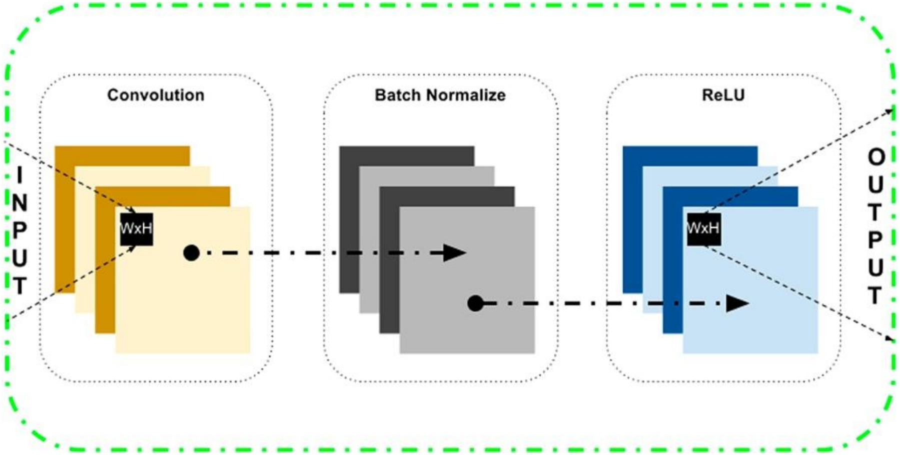


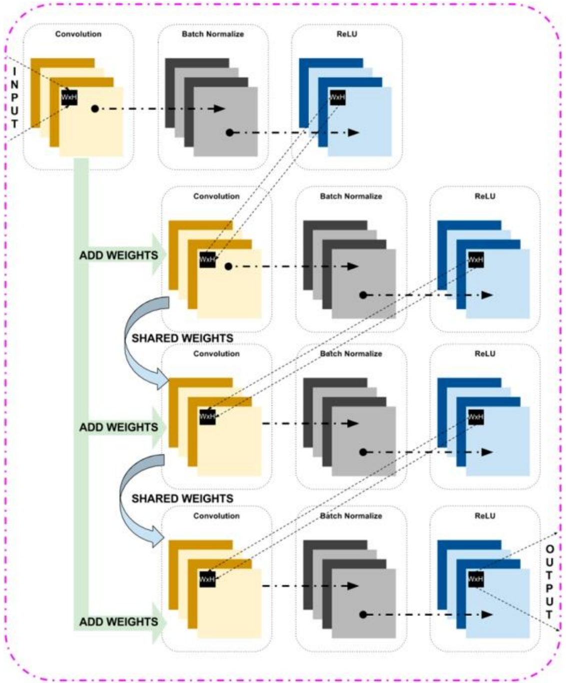
signal segments along the x, y, z acceleration axes

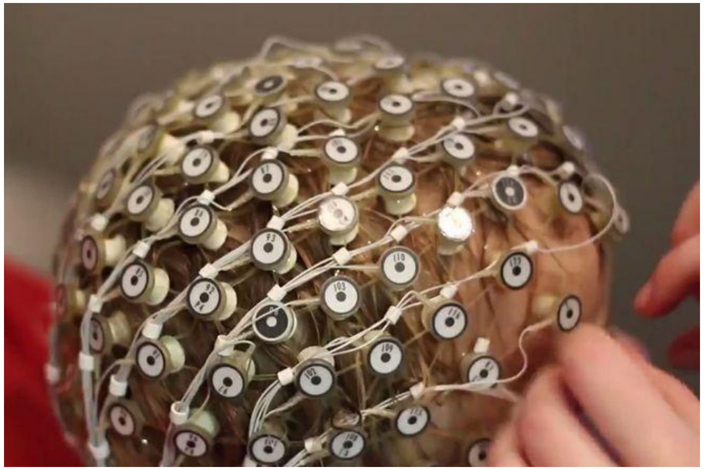
The cloudUPDRS app: A medical device for
the clinical assessment of Parkinson’s Disease
https://doi.org/10.1016/j.pmcj.2017.12.005



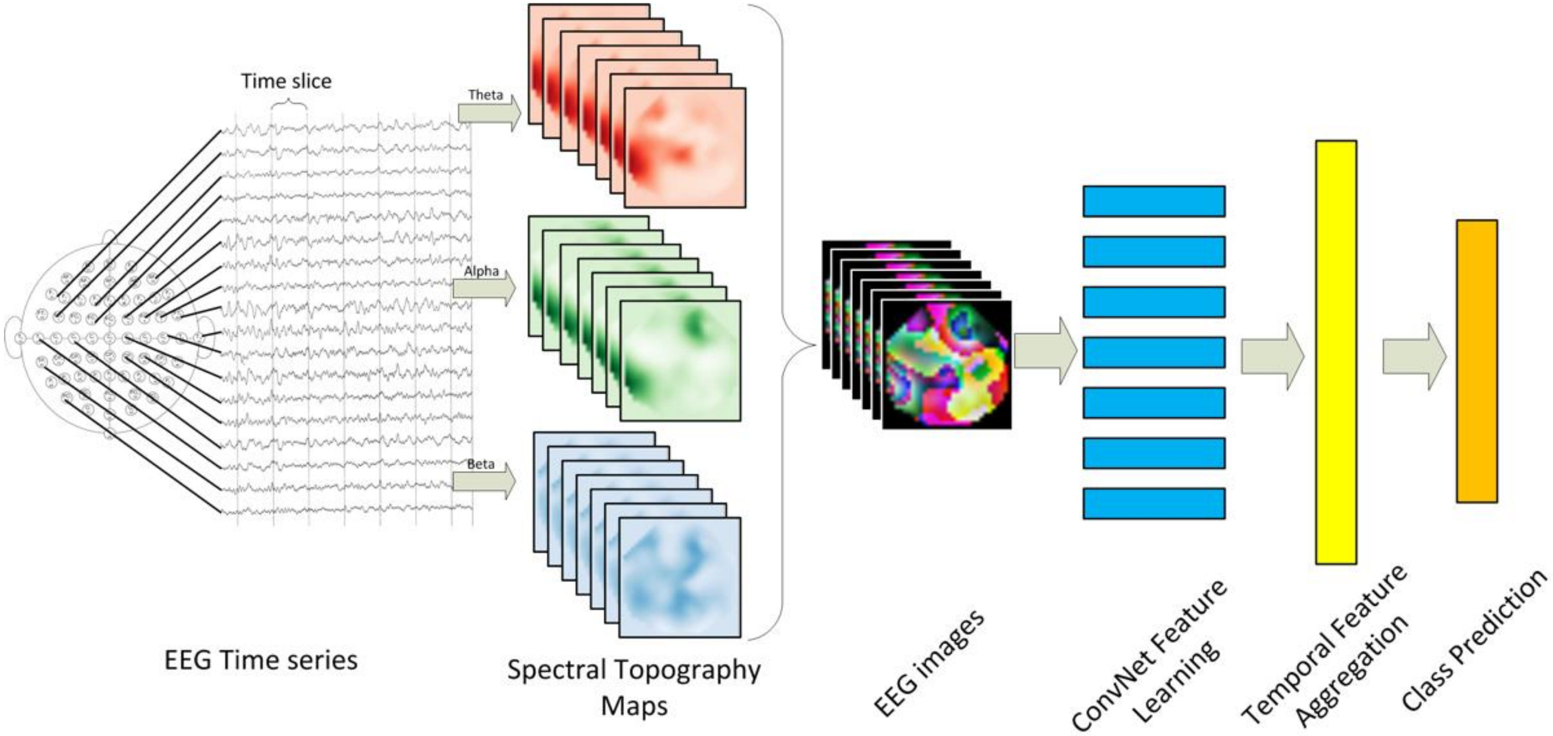
Compliance with protocol







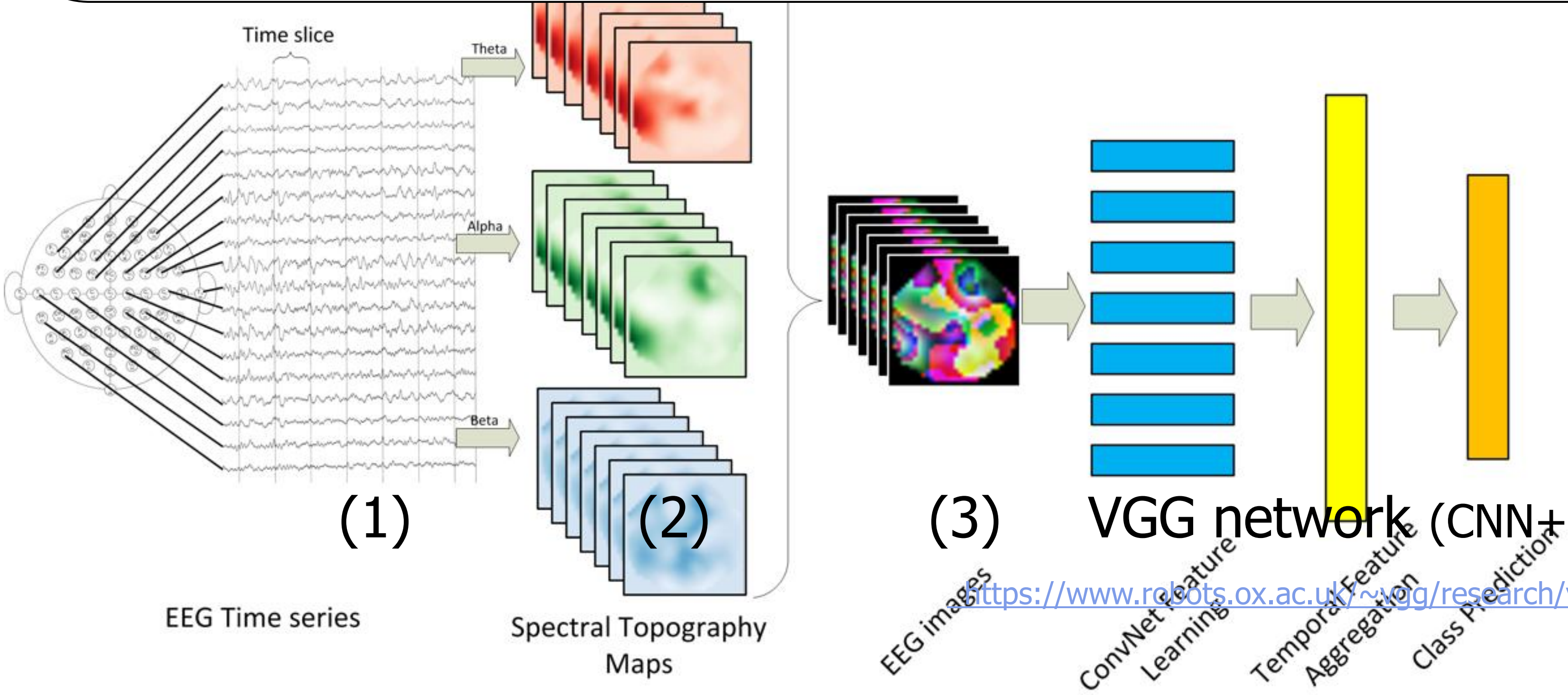
# Transform EEG activities into a sequence of topology-preserving multi-spectral images



**Cognitive load classification**

Learning Representations from EEG with Deep Recurrent-Convolutional Neural Networks- https://arxiv.org/abs/1511.06448

(1) EEG time series from multiple locations are acquired;
(2) spectral power within three prominent frequency bands is extracted for each location and used to form topographical maps for each time frame (Polar Projection, image);
(3) sequence of topographical maps are combined to form a sequence of 3-channel images which are fed into a recurrent-convolutional network



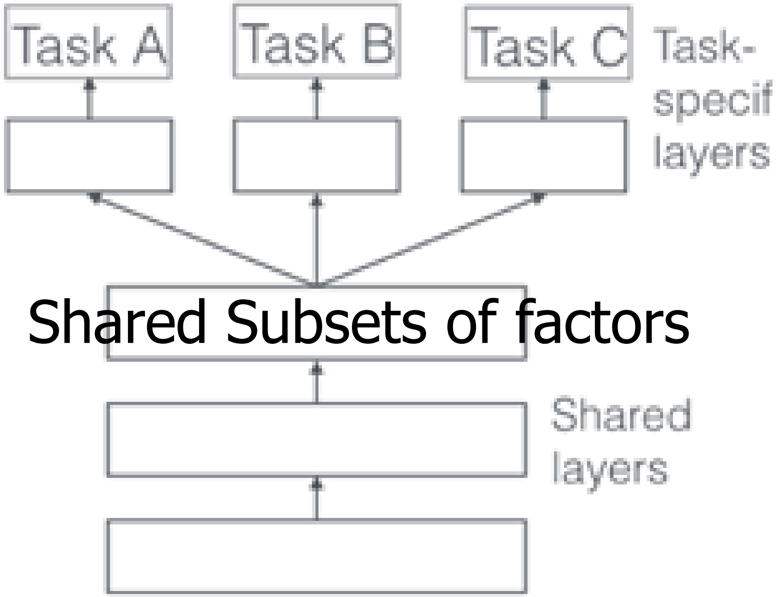
Cognitive load classification

(1) (2) (3) VGG network (CNN+LSTM)

https://www.robots.ox.ac.uk/~vgg/research/very_deep/

Learning Representations from EEG with Deep Recurrent-Convolutional Neural Networks- https://arxiv.org/abs/1511.06448

# Multi-task learning



Shared Subsets of factors

Input data

Advantageous in areas where it's natural to predict multiple related indicators simultaneously (e.g. finance, bioinformatics and drug discovery). It can reduce the risk of overfitting (regularization); increase the number of training data points (data augmentation); parallel tasks provide evidence for the relevance or irrelevance of different features; 'eavesdropping' across tasks (learn features G through task B whilst difficult to learn through A); shared representations can help the model perform well when learning novel tasks (as long as they are from the same environment).

# Summary

- Feature selection methods that are based on statistics. These can be used to assess how information rich are individual features to help machine learning methods discriminate between classes of objects or recognise objects.
- Class separability measures and ways to  combine features. These can be used to select good combinations of features to be used as input in machine learning methods
- Features learning using neural networks and deep learning architectures. Features are generated and selected as part of training process.

# Useful reading

- Theodoridis S., Koutroumbas K. (2009), chapter 5.1-5.4, 5.6.1, 5.6.3, 5.7.2, Pattern Recognition, Academic Press. Available online at: https://drive.google.com/file/d/0By995HEqDrWQbnBfTGJEVXZrRkE/view?usp=sharing

- Bengio Y., Courville A., Vincent P., Representation Learning: A Review and New Perspectives - https://arxiv.org/abs/1206.5538v3

# Next week

- Neural networks and deep learning