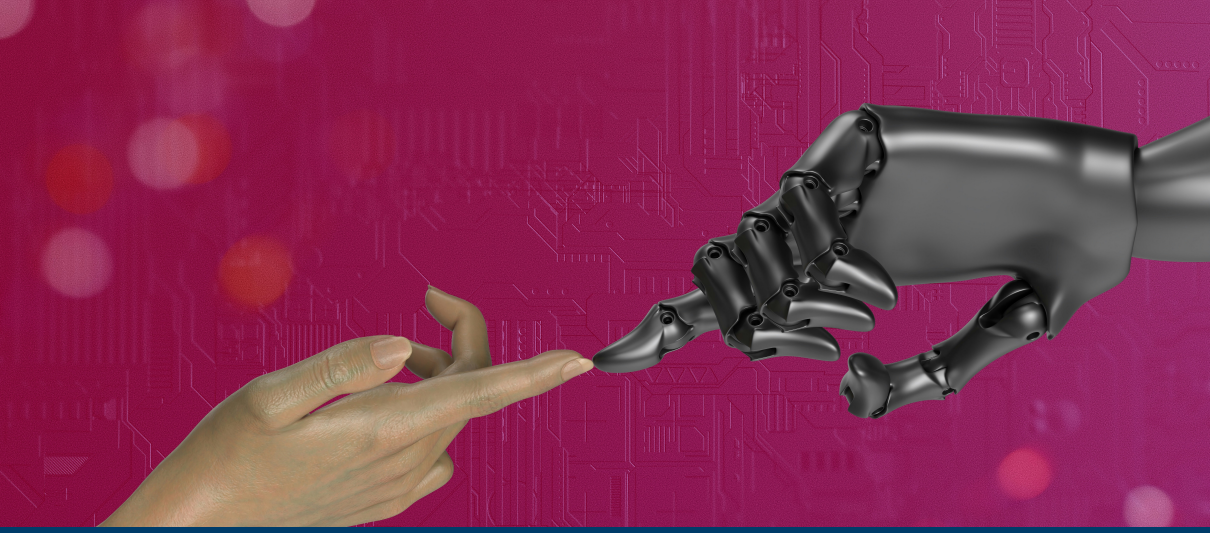


Hands-On Machine Learning

Dr. Derek Bridge | 2026/2027

UCC Coláiste na hOllscoile Corcaigh
University College Cork, Ireland

School of Computer Science and Information Technology
Scoil na Ríomheolaíochta agus na Teicneolaíochta Faisnéise

12. Feature Selection

In this lecture, we use a dataset about glass and its uses. It was collected by the *USA Forensic Science Service*. It is publicy-available from the UCI Machine Learning Repository.

Recap

- **Overfitting**: the model is too complex.
- One remedy, especially, for parametric models, is to reduce the number of features (and hence the number of parameters).
- Regularization of a parametric model using the $l_1$norm (Lasso) can set weights to zero, which is a form of feature selection.

Feature Selection and Dimensionality Reduction

- **Feature Selection**: select a subset of the features.
  - Sequential Feature Selection Algorithms.
  - Filter Methods that use a scoring function.
  - Filter Methods that use a separate model.
- **Dimensionality Reduction**: map the existing features to a new feature space and retain only some of the new features.

Sequential Feature Selection Algorithms

- There are *many* approaches to feature selection.
- Among the simplest are **Sequential Feature Selection Algorithms**.
- Greedy algorithms choosing a feature to remove/add one at a time.
- Stop when you have the desired number of features — a hyperparameter, so could choose it by, e.g., grid search.
- But, if there are many features, then these algorithms train and evaluate many models — expensive!

Sequential Backward Selection

$\mathcal{F} \leftarrow$ all the features;

repeat
  Determine which feature f has least impact on validation error;

  Remove f from $\mathcal{F}$;

until *until stopping criterion*;

Sequential Forward Selection

$\mathcal{F} \leftarrow \{\}$;

repeat
  Determine which feature f has most impact on validation error;

  Insert f into $\mathcal{F}$;

until *until stopping criterion*;

Filter Methods for Feature Selection

- Sequential Feature Selection Algorithms are costly to run.
- The alternative is to use a **filter method**.
- In a filter method, we compute **feature importances** in some separate way.
- Then we retain the features with highest importance.

Examples

- There are numerous filter methods available.
- For regression, feature importance can be the F-value — based on the Pearson correlation between the feature and the target.
- For classification, feature importance can be the ANOVA F-value — a version of the F-value adapted for classification.

Disadvantages

- Being based on Pearson, these scores measure linear relationships only.
- This approach measures importance for each feature in isolation — ignoring whether they are redundant when other features are present or more useful jointly with other features.

Using a Separate Model

- Some ML models can output feature importances, e.g. Decision Trees, Random Forests.
- Instead of using, e.g., F-values, we can train a Decision Tree or Random Forest, use its feature importances to filter the features, then train the model we ultimately want.
- This can overcome the disadvantages above.
- It may be more accurate than the Decision Tree or Random Forest on its own.

Principal Components Analysis

Principal Components Analysis (PCA) gives a method for **dimensionality reduction**.

- PCA creates new features —called **components**— based on the existing ones. The components are linear combinations of the existing features. There is one component per existing feature.
- But these components are ordered — meaning we can discard the least important ones.

Mathematically speaking...

- PCA mean-centers the training examples — by subtracting each feature's mean from the feature-values.
- Calculates the covariance matrix of the mean-centered training examples.
- Calculates the eigenvectors of the covariance matrix. These are the axes of a transformed space — the new features. But there are still n of them.
- Retains a subset of the new features — the ones with the largest eigenvalues.

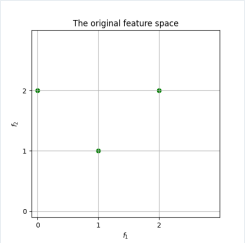
There follows a very informal presentation of PCA.

Example

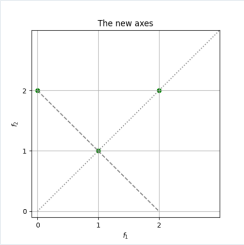
Suppose a dataset describes objects using just two features, f_1 and f_2.

f_1	f_2
1	1
2	2
0	2

Since there are only two features, we can plot — a 2D visualization:



- But the co-ordinate system we use is arbitrary.
- A different pair of axes would work just as well.
- E.g. we could draw one axis diagonally.
- By convention, the other will be perpendicular (or orthogonal) to the first — in other words, at right angles.
- But, even then, we can decide where along the first axis to place it.



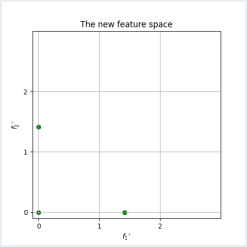
- On this new 'graph paper', the co-ordinates of the examples are now different.

f_1	f_2
1	1
2	2
0	2

becomes

f_1'	f_2'
0	0
$\sqrt{2}$	0
0	$\sqrt{2}$

- We can see this if we rotate the graph paper, so that the diagonal is horizontal:



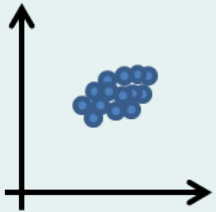
- These new features convey exactly the same information as the originals — they're just on a different co-ordinate system.

Choosing the new axes

- There is an infinite choice of axes!
- In PCA, the orientation of the first axis is based on covering the greatest variance (spread).
- Each subsequent axis must cover as much remaining spread as possible but with the constraint that they must be perpendicular to one another.
- Making these decisions does require that the features be comparable. This is why PCA mean-centers their values first.

Question

The plot contains examples described by two features, which you can assume have been standardized.



PCA will choose two new axes. Where will they be?

Principal Components

- At this stage, we've gained very little: we had n features (axes), we still have n features (new axes, components).
- But examples in real-world datasets often lie close to a lower-dimensional subspace of the high-dimensional space.
- So, we can re-express our data ('project it') to just the first n' axes ($n' < n$), hopefully, with very little loss of information.

Explained Variance Ratio

- How do we choose n' — the number of components to retain?
- We can treat n' as a hyperparameter and set it using, e.g., grid search.
- Or we can plot the **explained variance ratio** of each component — tells us how much of the training sets' variance lies along that axis.
- There is often an 'elbow' in the plot — this gives the value of n'

Limitations of PCA

- It can only be used on the numeric-valued features.
- Its new features are always linear combinations of the existing ones: in essence, we're fitting straight-line axes through the values.
- Being a form of unsupervised learning, it ignores the target values/labels. (This is also an advantage: it can be used for both regression and classification.)

Alternatives to PCA

- There are many techniques that work in a similar way but with more flexibility.
- For non-numeric data, e.g. Canonical Correspondence Analysis (CCA).
- For bag-of-words (next lecture), e.g. Singular Value Decomposition (SVD).
- To allow non-linear combination, e.g. Kernel PCA, Isomap, Locally Linear Embedding, Multidimensional Scaling (MDS).
- To take the classes into account in classification tasks, e.g. Linear Discriminant Analysis (LDA).