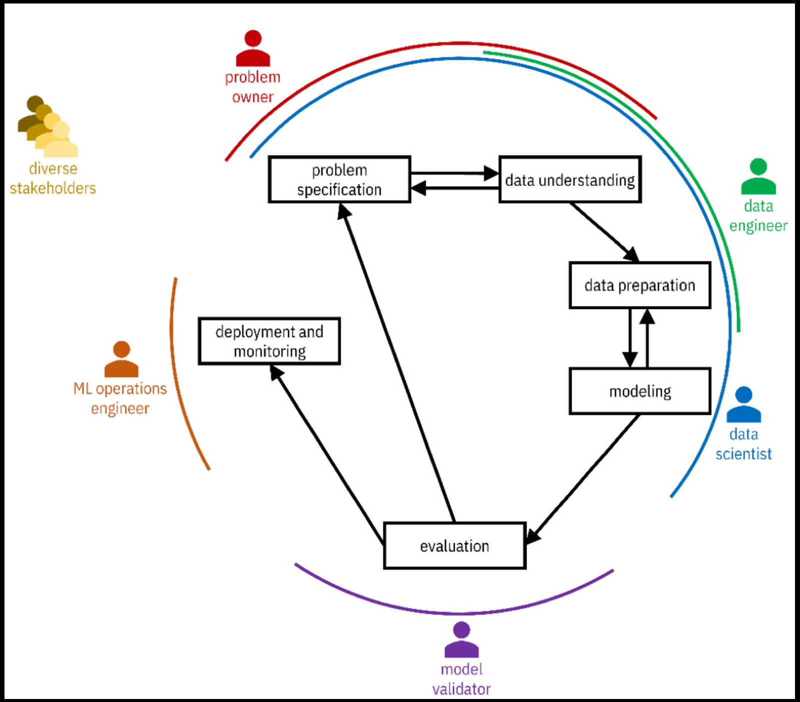
Machine Learning for Design

Lecture 7

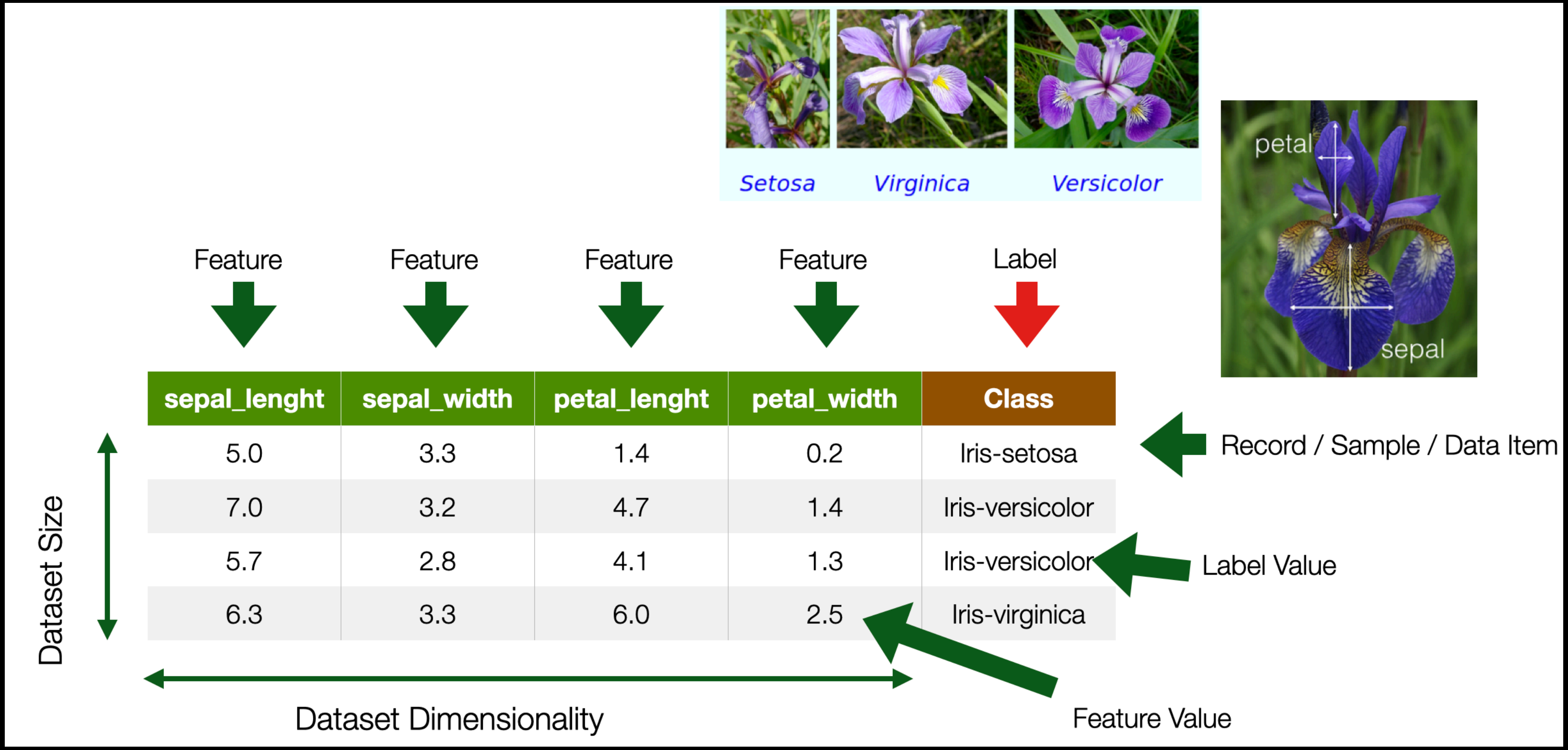
Design and Develop Machine Learning Models - *Part 1*

Previously on ML4D

CRISP-DM Methodology



Data



sepal_lenght	sepal_width	petal_lenght	petal_width	Class
5.0	3.3	1.4	0.2	Iris-setosa
7.0	3.2	4.7	1.4	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-versicolor
6.3	3.3	6.0	2.5	Iris-virginica

Types of Features / Label Values

- **Categorical**
  - Named Data
  - Can take numerical values, but no mathematical meaning
- **Numerical**
  - -Measurements
  - Take numerical values (discrete or continuous)

Categorical Nominal

- No order
- No direction
- e.g. marital status, gender, ethnicity

Categorical Ordinal

- Order
- Direction
- e.g., letter grades (*A*,*B*,*C*,*D*), ratings (*dislike*, *neutral*, *like*)

Numerical Interval

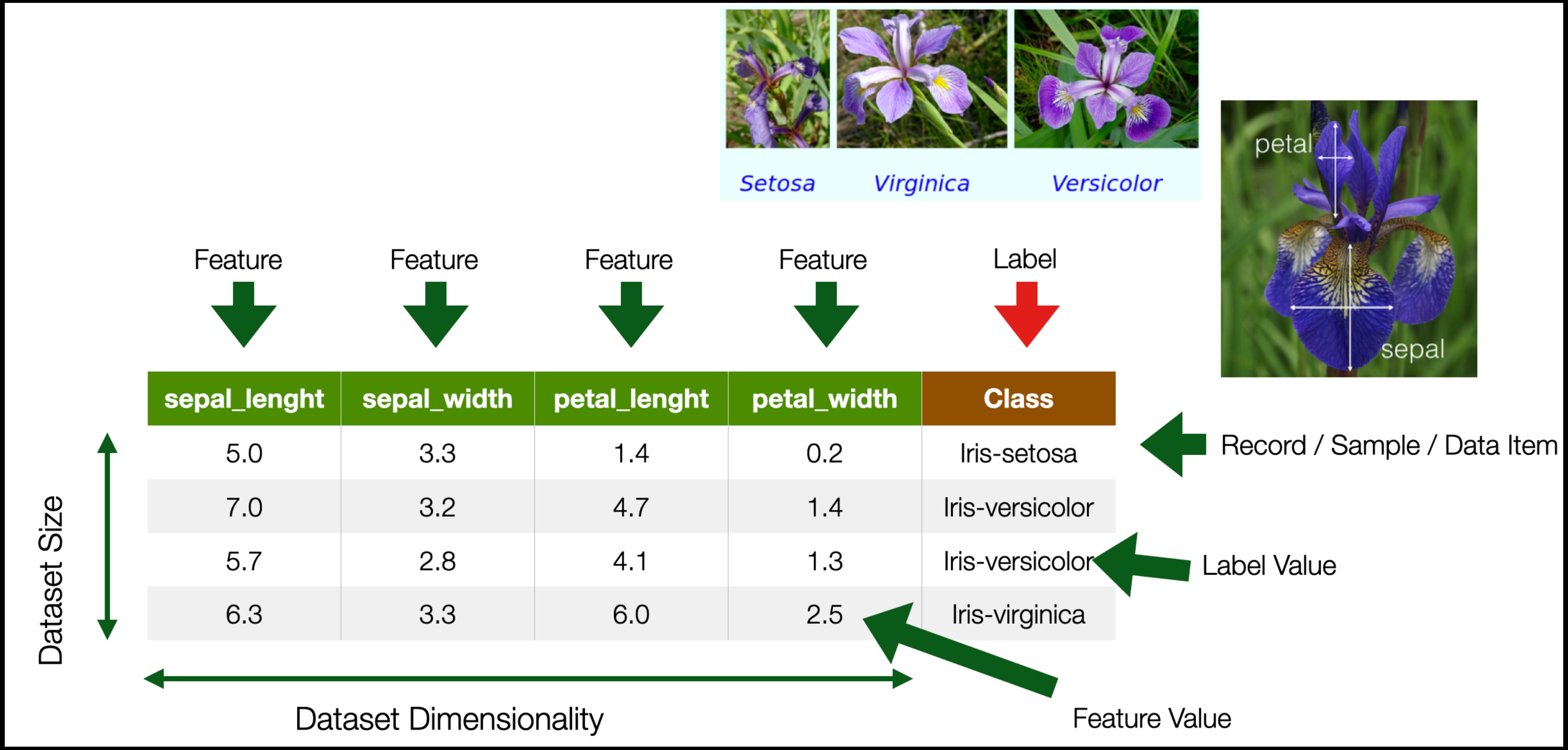
- Difference between measurements
- No true zero or fixed beginning
- e.g., temperature (C or F), IQ, time, dates

Numerical Ratio

- Difference between measurements
- True zero exists
- e.g., temperature (K), age, height

Data Preparation

Ideal Data



Feature Feature Feature Feature Label

sepal_lenght	sepal_width	petal_lenght	petal_width	Class
5.0	3.3	1.4	0.2	Iris-setosa
7.0	3.2	4.7	1.4	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-versicolor
6.3	3.3	6.0	2.5	Iris-virginica

Record / Sample / Data Item

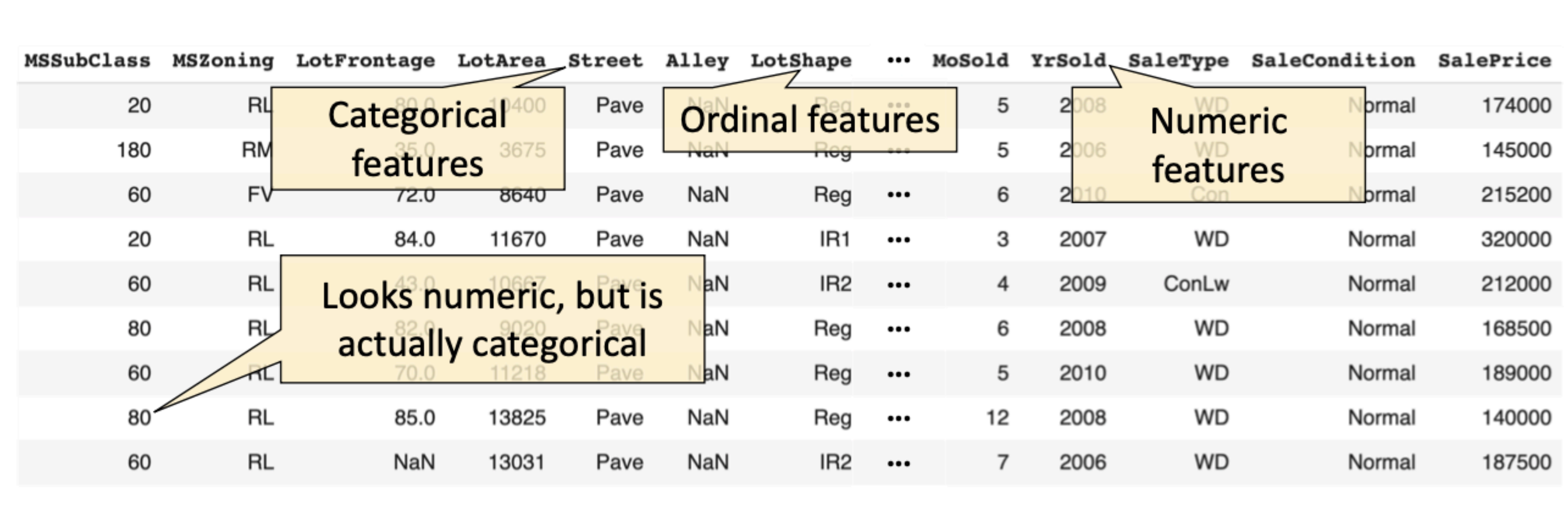
Label Value

Dataset Size

Dataset Dimensionality

Feature Value

Real Data



MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	•••	MoSold	YrSold	SaleType	SaleCondition	SalePrice
20	RL	80.0	10400	Pave	NaN	Reg	•••	5	2008	WD	Normal	174000
180	RM	35.0	3675	Pave	NaN	Reg	•••	5	2006	WD	Normal	145000
60	FV	72.0	8640	Pave	NaN	Reg	•••	6	2010	Con	Normal	215200
20	RL	84.0	11670	Pave	NaN	IR1	•••	3	2007	WD	Normal	320000
60	RL	43.0	10667	Pave	NaN	IR2	•••	4	2009	ConLw	Normal	212000
80	RL	82.0	9020	Pave	NaN	Reg	•••	6	2008	WD	Normal	168500
60	RL	70.0	11218	Pave	NaN	Reg	•••	5	2010	WD	Normal	189000
80	RL	85.0	13825	Pave	NaN	Reg	•••	12	2008	WD	Normal	140000
60	RL	NaN	13031	Pave	NaN	IR2	•••	7	2006	WD	Normal	187500

Categorical features

Ordinal features

Numeric features

Looks numeric, but is actually categorical

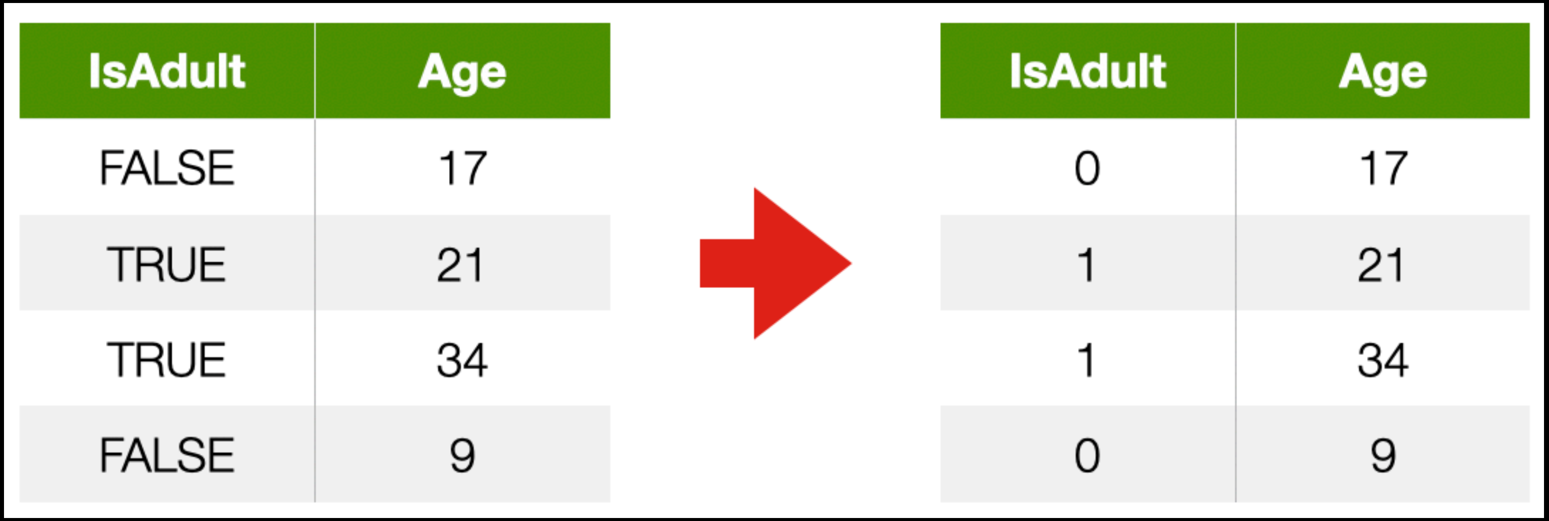
- Data is rarely “clean”
- Approximately 50-80% of the time is spent on data wrangling
  - probably an under-estimation
- Having good data with the correct features is critical

- 3 issues to deal with:
  - **Encoding** features as numerical values
  - **Transforming** features to make ML algorithms work better
  - Dealing with **missing feature** values

Data Encoding

Numerical Features

– Each feature is assigned its own value in the feature space



IsAdult	Age
FALSE	17
TRUE	21
TRUE	34
FALSE	9

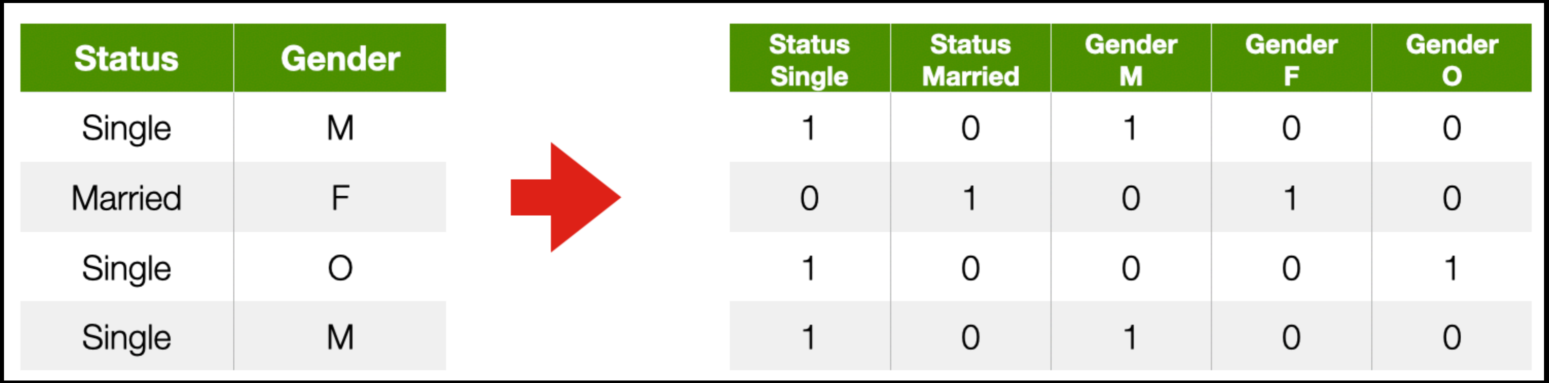
IsAdult	Age
0	17
1	21
1	34
0	9

Categorical Features

- Why not encode each value as an integer?
  - A naive integer encoding would create an ordering of the feature values that *does not exist in the original data*
  - You can try direct integer encoding if a feature *does have a natural ordering* (ORDINAL e.g. ECTS grades A–F)

One-hot Encoding

- Each value of a categorical feature gets its own column

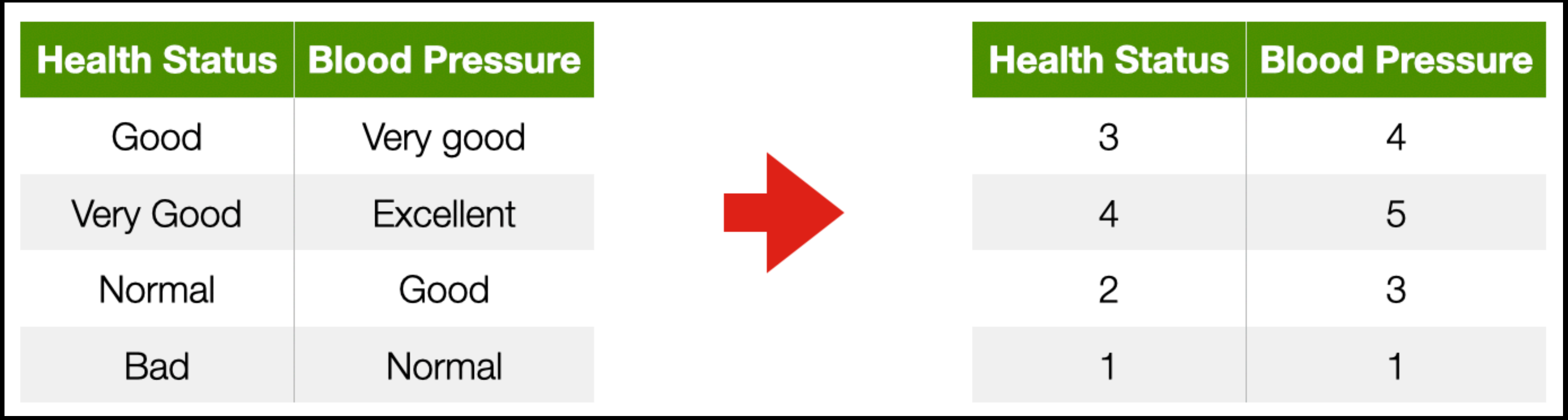


Status	Gender
Single	M
Married	F
Single	O
Single	M

Status Single	Status Married	Gender M	Gender F	Gender O
1	0	1	0	0
0	1	0	1	0
1	0	0	0	1
1	0	1	0	0

Ordinal Features

- Convert to a number, preserving the order
  - $[low, medium, high] \rightarrow [1, 2, 3]$
- **Encoding may not capture relative differences**



Health Status	Blood Pressure
Good	Very good
Very Good	Excellent
Normal	Good
Bad	Normal

Health Status	Blood Pressure
3	4
4	5
2	3
1	1

Data Quality Issues

Incorrect feature values

- Typos
  - e.g., color = $``blue", ``green", ``gren", ``red"$
- Garbage
  - e.g., color = “w☰r--śï¿”
- Inconsistent spelling (e.g., “color”, “colour”) or capitalization
- Inconsistent abbreviations (e.g., “Oak St.”, “Oak Street”)

Missing labels (classes)

- Delete instances if only a few are missing labels
- Use semi-supervised learning techniques
- Predict the missing labels via self-supervision

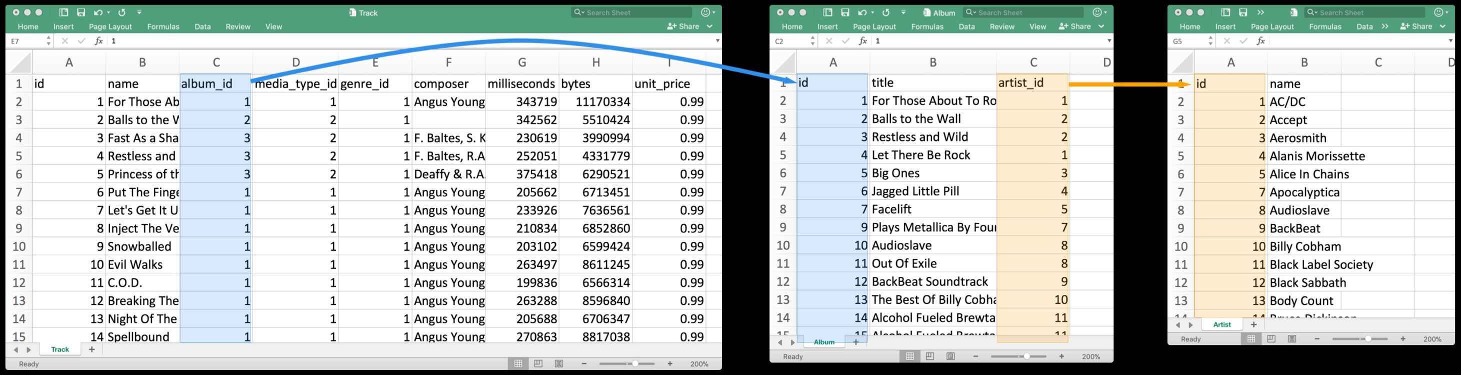
Merging Data

- Data may be split across different files (or systems!)
  - *join* based on a key to combine data into one table

tracks

albums

artists



	A	B	C	D	E	F	G	H	I
1	id	name	album_id	media_type_id	genre_id	composer	milliseconds	bytes	unit_price
2	1	For Those Ab	1	1	1	Angus Young	343719	11170334	0.99
3	2	Balls to the V	2	2	1		342562	5510424	0.99
4	3	Fast As a Sha	3	2	1	F. Baltes, S. K	230619	3990994	0.99
5	4	Restless and	3	2	1	F. Baltes, R.A	252051	4331779	0.99
6	5	Princess of th	3	2	1	Deaffy & R.A.	375418	6290521	0.99
7	6	Put The Finge	1	1	1	Angus Young	205662	6713451	0.99
8	7	Let's Get It U	1	1	1	Angus Young	233926	7636561	0.99
9	8	Inject The Ve	1	1	1	Angus Young	210834	6852860	0.99
10	9	Snowballed	1	1	1	Angus Young	203102	6599424	0.99
11	10	Evil Walks	1	1	1	Angus Young	263497	8611245	0.99
12	11	C.O.D.	1	1	1	Angus Young	199836	6566314	0.99
13	12	Breaking The	1	1	1	Angus Young	263288	8596840	0.99
14	13	Night Of The	1	1	1	Angus Young	205688	6706347	0.99
15	14	Spellbound	1	1	1	Angus Young	270863	8817038	0.99

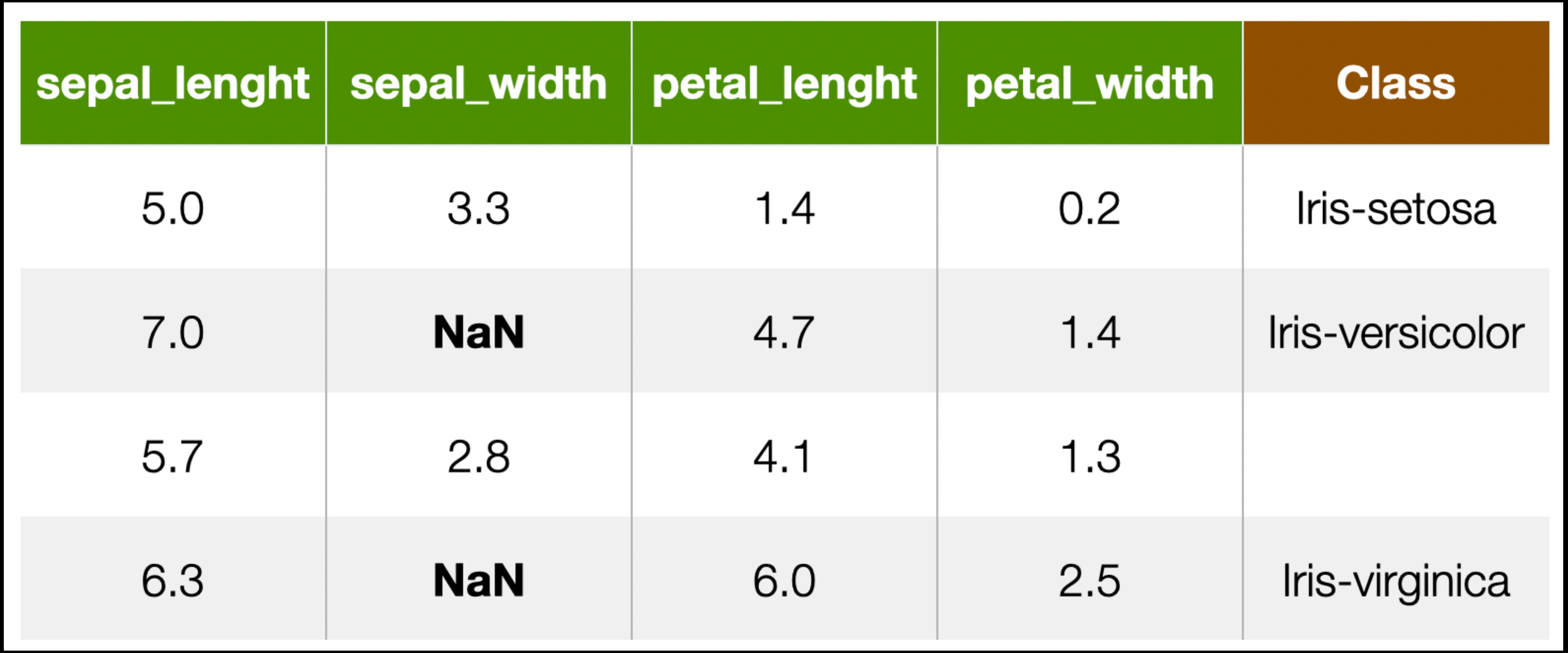
	A	B	C
1	id	title	artist_id
2	1	For Those About To Ro	1
3	2	Balls to the Wall	2
4	3	Restless and Wild	2
5	4	Let There Be Rock	1
6	5	Big Ones	3
7	6	Jagged Little Pill	4
8	7	Facelift	5
9	9	Plays Metallica By Fou	7
10	10	Audioslave	8
11	11	Out Of Exile	8
12	12	BackBeat Soundtrack	9
13	13	The Best Of Billy Cobha	10
14	14	Alcohol Fueled Brewta	11
15	15	Alcohol Fueled Brewta	11

	A	B
1	id	name
2	1	AC/DC
3	2	Accept
4	3	Aerosmith
5	4	Alanis Morissette
6	5	Alice In Chains
7	7	Apocalyptica
8	8	Audioslave
9	9	BackBeat
10	10	Billy Cobham
11	11	Black Label Society
12	12	Black Sabbath
13	13	Body Count
14	14	Bruce Dickinson

Problems During Merge

- Inconsistent data
  - Same instance key with conflicting labels
  - Data duplication
- Data size
  - Data might be too big to integrate
- Encoding issues
  - Inconsistent data formats or terminology
  - Key aspects mentioned in cell comments or auxiliary files

Dealing With Missing Values



sepal_lenght	sepal_width	petal_lenght	petal_width	Class
5.0	3.3	1.4	0.2	Iris-setosa
7.0	**NaN**	4.7	1.4	Iris-versicolor
5.7	2.8	4.1	1.3	
6.3	**NaN**	6.0	2.5	Iris-virginica

Why can data be missing?

- "Good" reason: not all instances are meant to have a value
- Otherwise
  - Technical issues (e.g. Data Quality)

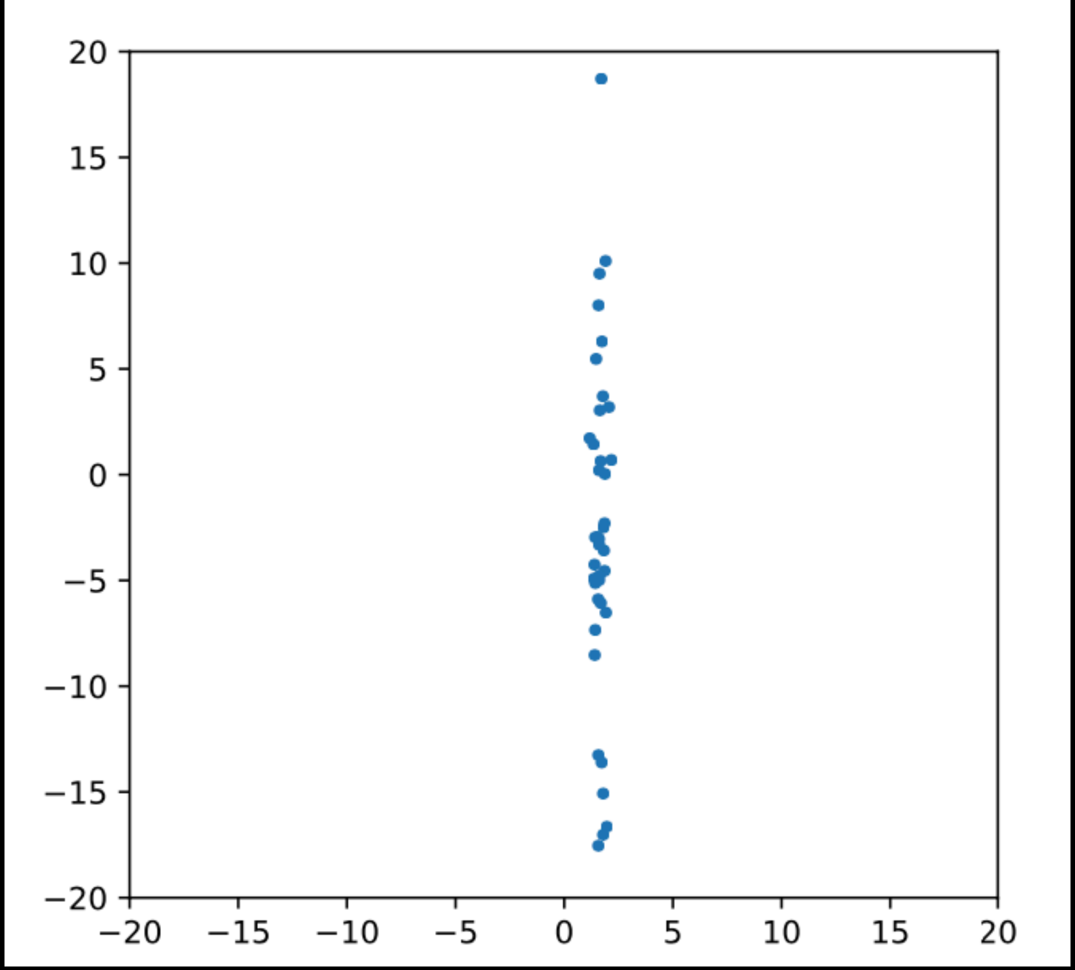
Dealing with missing data

- **Delete features** with mostly missing values (columns)
- **Delete instances** with missing features (rows)
  - Only if rare
- **Feature imputation**
  - “fill in the blanks"

Feature Imputation

- **Replacing** with a **constant**
  - the *mean* feature value (numerical)
  - the *mode* (categorical or ordinal)
  - “flag” missing values using out-of-range values
- **Replacing** with a **random** value
- **Predicting** the feature value **from other features**

What if our features look like this?



- What if the features have different magnitudes?
- Does it matter if a feature is represented as meters or millimetres?
- What if there are outliers?

- Values spread strongly affect many models:
  - linear models (linear SVC, logistic regression, . . .)
  - neural networks
  - models based on distance or similarity (e.g. kNN)
- It does not matter for most tree-based predictors

Feature Normalisation

- Needed for many algorithms to work properly
    - Or to speed up training

Min/Max Scaling

$$f_{new} = \frac{f - f_{max}}{f_{max} - f_{min}}$$

- Values scaled between 0 and 1
- f_{max} and f_{min} need to be known in advance

Standard Scaling

$$f_{new} = \frac{f - \mu_f}{\sigma_f}$$

- Rescales features to have zero mean and unit variance
- Outliers can cause problems

Scaling to unit length

$$x_{new} = \frac{x}{|x|}$$

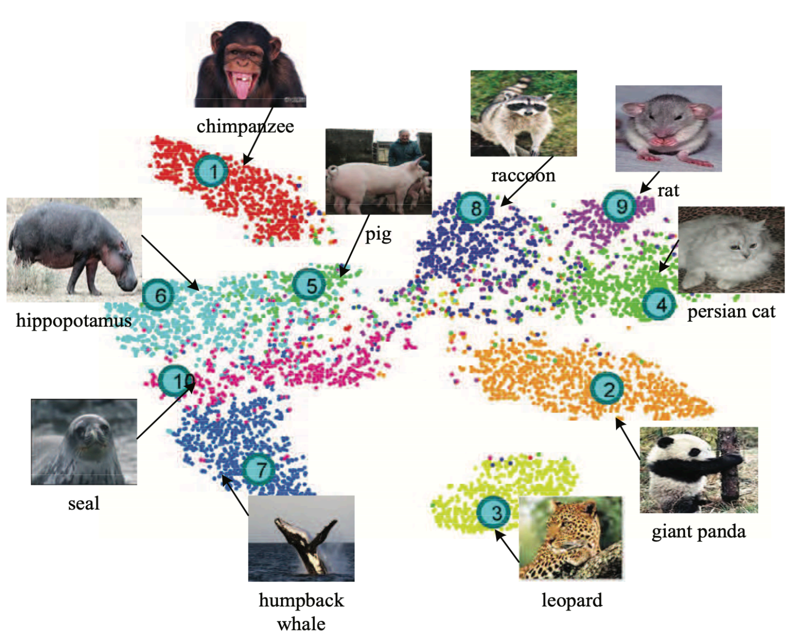
- Typical for textual document

Other features transformation

- Improve performance by applying other numerical transformation
  - logarithm, square root, . . .
  - TF-IDF
- It depends a lot on the data!
  - Trial and error
  - Exploration
  - Intuition

Feature Selection and Removal

- **Problem**: the number of features may be very large
  - Important information is drowned out
  - Longer model training time
  - More complexity → bad for generalization
- **Solution**: leave out some features
  - But which ones?
- **Feature selection methods** can find a useful subset

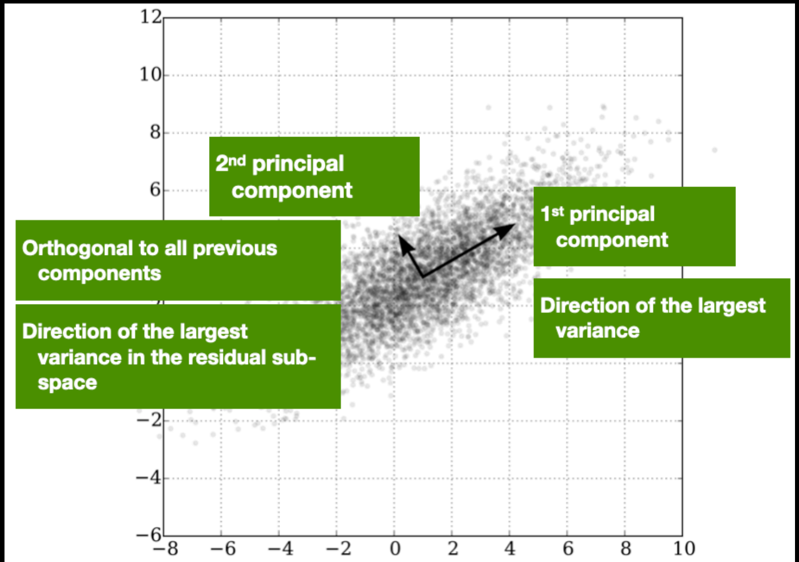


Feature Selection

- **Idea**: find a subspace that retains most of the information about the original data
  - Pretty much as we were doing with *word embeddings*
- **PRO**: fewer dimensions make for datasets that are easier to explore and visualise, and faster training of ML algorithms
- **CONS**: drop in prediction accuracy (less information)
- There are many different methods, **Principal Component Analysis** is a classic

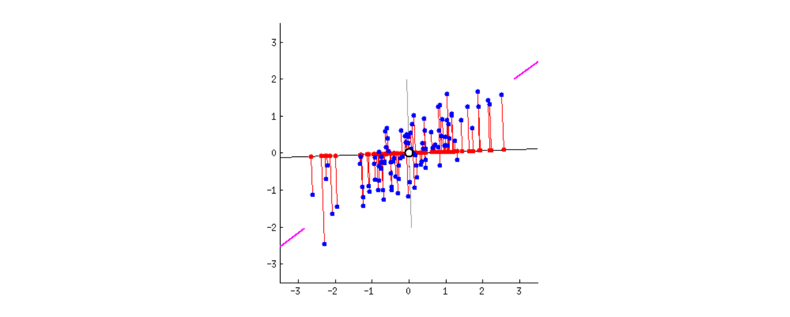
Principal Component Analysis

- **Idea**: features can be highly correlated with each other
  - redundant information
- **Principal components**: new features constructed as *linear combinations* or *mixtures* of the initial features
- The new features (i.e., principal components) are **uncorrelated**
  - Most of the information within the initial features is compressed into the first components



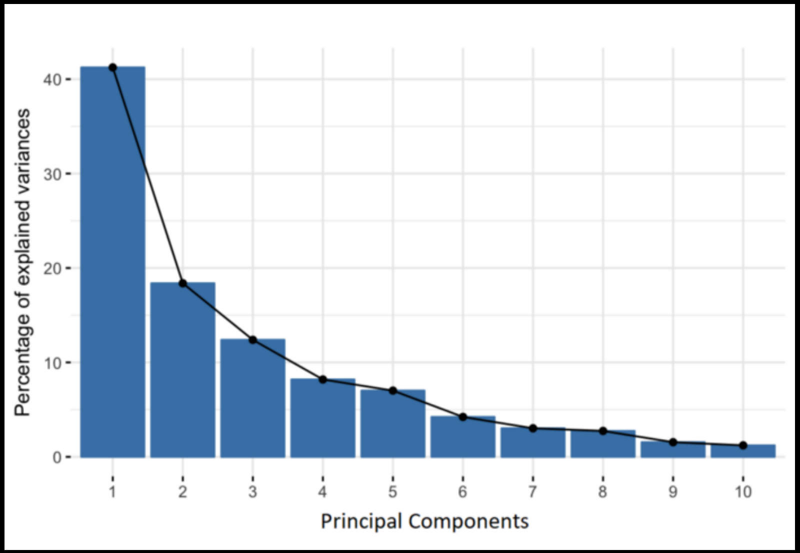
Principal Component Analysis

- Orthogonal projection of data onto lower-dimension linear space that:
  - *Maximizes the variance* of projected data (purple line)
  - *Minimizes mean squared distance* between data point and projections (sum of red lines)



Dimensionality Reduction

- **Use** the PCA transformation of the data instead of the original features
- **Ignore** the components of less significance (e.g., only pick the first three components)



- PCA keeps most of the variance of the data
- So, we are reducing the dataset to features that retain meaningful variations of the dataset

And now, let's Smell Pittsburgh

Credits: Yen-Chia Hsu

Machine Learning for Design

Lecture 7

Design and Develop Machine Learning Models - *Part 1*

Credits

CIS 419/519 Applied Machine Learning. Eric Eaton, Dinesh Jayaraman.

A Step-by-Step Explanation of Principal Component Analysis (PCA).