Python for Data Science and Machine Learning

School Year 2023-2024

IST

Course Structure

1	Introduction to Python	OCT 19, OCT 26, NOV 9, NOV 16, NOV 23, NOV 30, DEC 7
2	Python Challenge	DEC 14
3	Introduction to Data Science	DEC 21, JAN 11, JAN 18, JAN 25
4	Introduction to Machine Learning	FEB 1, FEB 22, FEB 29
5	Data Science & ML Challenge	MAR 7

= Core Topics   = Optional Topics

Jupyter Notebook Setup

In a browser:

```
192.168.10.4:8888
```

Password: **ist**

Recap: Pandas & other LIbraries

Pandas is a powerful Python data analysis toolkit.

Matplotlib & **Seaborn** are plotting libraries.

14.0

```
import pandas as pd
import numpy as np
```

I have added functions (**plot_2d** & **plot_3d**, etc) that will help plotting charts in future exercises

Recap: DataFrame

A **DataFrame** is a two-dimensional data structure with labeled axes (rows and columns).

14.1

```
df = pd.read_csv("titanic_dataset.csv")
df
```

Recap: DataFrame

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

Recap: Exploratory Data Analysis (EDA)

Before we dive into Machine Learning: EDA!

Exploratory Data Analysis refers to the critical process of performing initial **investigations on data** so as to discover **patterns**,to spot **anomalies**,to test hypothesis and to check **assumptions**.

Pratil, Prasad. (2018). "What is Exploratory Data Analysis?" Towards Data Science.
Available at: https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15.

Recap: Feature Engineering

Feature engineering or feature extraction or feature discovery is the process of **extracting features** (characteristics, properties, attributes) **from raw** data **to support training** a downstream statistical model.

Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome H. (2009).

The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer. ISBN 978-0-387-84884-6.

Recap: Analysing the “Embarked” Column

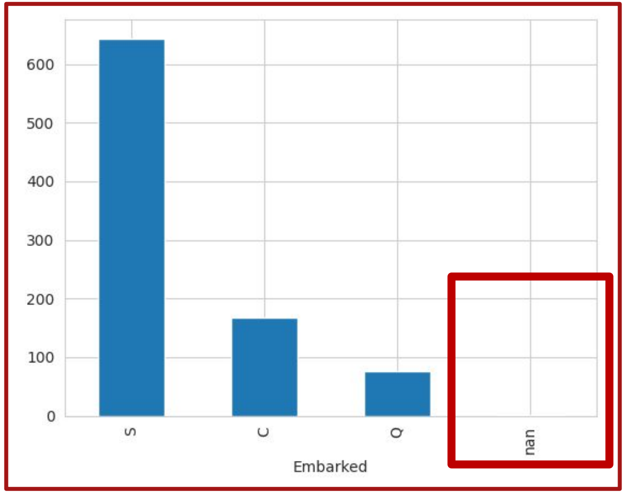
We can see that not all passengers have data regarding their embarkation point:

```
df[pd.isna(df["Embarked"])]
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
61	62	1	1	Icard, Miss. Amelie	female	38.0	0	0	113572	80.0	B28	NaN
829	830	1	1	Stone, Mrs. George Nelson (Martha Evelyn)	female	62.0	0	0	113572	80.0	B28	NaN

Recap: Analysing the "Embarked" Column

To visualise the current value distribution:



Recap: Analysing the “Age” Column

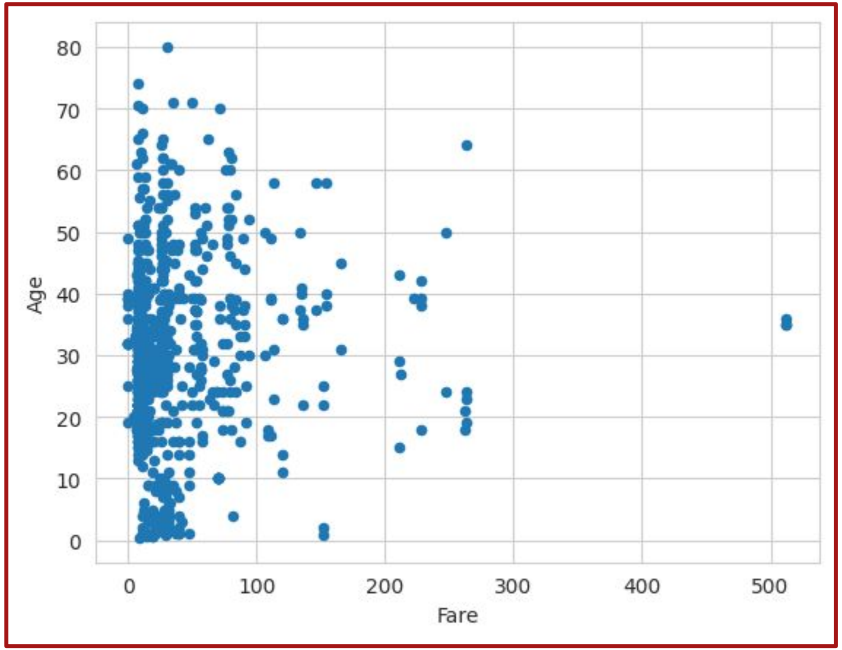
We can see that not all passengers have data on their age:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
17	18	1	2	Williams, Mr. Charles Eugene	male	NaN	0	0	244373	13.0000	NaN	S
19	20	1	3	Masselmani, Mrs. Fatima	female	NaN	0	0	2649	7.2250	NaN	C
26	27	0	3	Emir, Mr. Farred Chehab	male	NaN	0	0	2631	7.2250	NaN	C
28	29	1	3	O'Dwyer, Miss. Ellen "Nellie"	female	NaN	0	0	330959	7.8792	NaN	Q
...	...	...	...	...	...	...	...	...	...	...	...	...
859	860	0	3	Razi, Mr. Raihed	male	NaN	0	0	2629	7.2292	NaN	C
863	864	0	3	Sage, Miss. Dorothy Edith "Dolly"	female	NaN	8	2	CA. 2343	69.5500	NaN	S
868	869	0	3	van Melkebeke, Mr. Philemon	male	NaN	0	0	345777	9.5000	NaN	S
878	879	0	3	Laleff, Mr. Kristo	male	NaN	0	0	349217	7.8958	NaN	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S

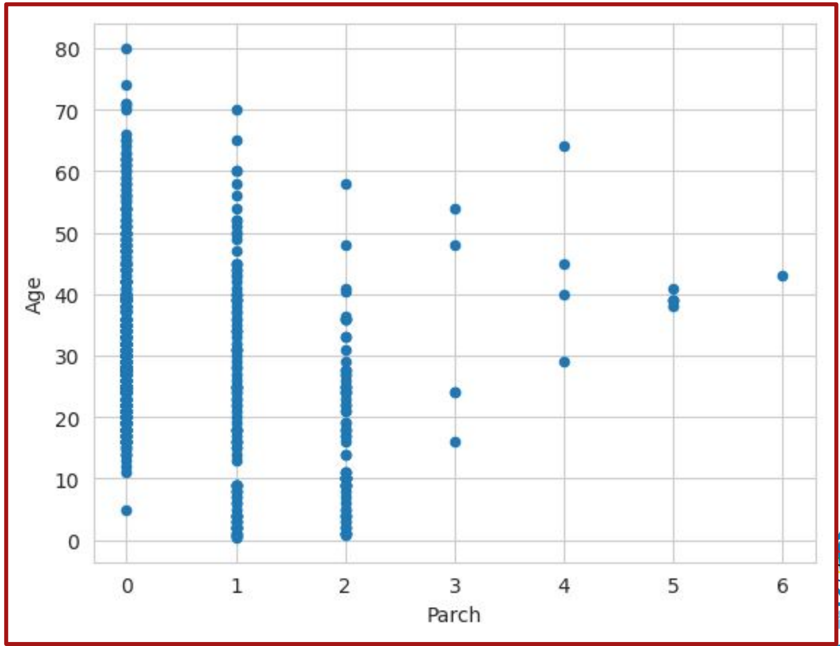
177 rows × 12 columns

Recap: Analysing the "Age" Column

Is **"Fare"** correlated?

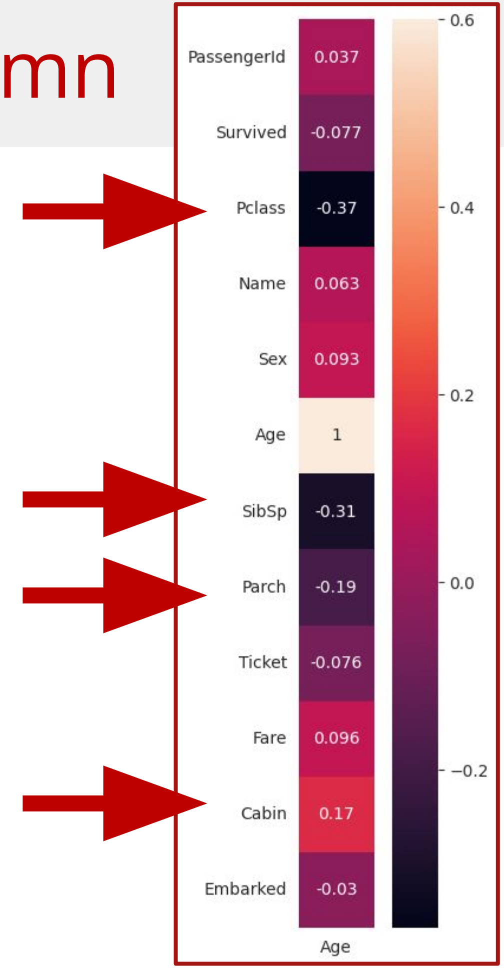


Is **"Parch"** correlated?



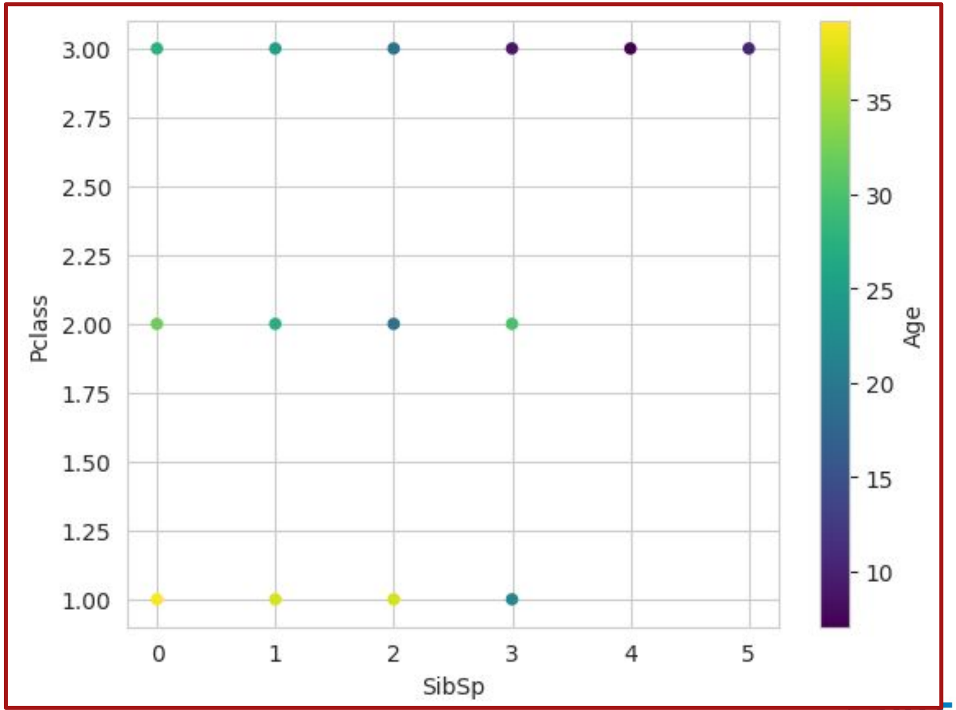
Recap: Analysing the “Age” Column

We can actually view the correlations across all columns in the dataframe:



Recap: Analysing the “Age” Column

Let’s visualise how **Pclass** and **SibSp** changes affect the average Age value:



You can see the raw numbers:

Recap: Categorization

Data Science does **not** work well with strings.

Categorization is the act of mapping **strings to ints/floats**.

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	CatSex	CatEmbarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.000000	1	0	A/5 21171	7.2500	NaN	S	0	0
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.000000	1	0	PC 17599	71.2833	C85	C	1	1
2	3	1	3	Heikkinen, Miss. Laina	female	26.000000	0	0	STON/O2. 3101282	7.9250	NaN	S	1	0
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.000000	1	0	113803	53.1000	C123	S	1	0
4	5	0	3	Allen, Mr. William Henry	male	35.000000	0	0	373450	8.0500	NaN	S	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.000000	0	0	211536	13.0000	NaN	S	0	0
887	888	1	1	Graham, Miss. Margaret Edith	female	19.000000	0	0	112053	30.0000	B42	S	1	0
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	24.912698	1	2	W./C. 6607	23.4500	NaN	S	1	0
889	890	1	1	Behr, Mr. Karl Howell	male	26.000000	0	0	111369	30.0000	C148	C	0	1
890	891	0	3	Dooley, Mr. Patrick	male	32.000000	0	0	370376	7.7500	NaN	Q	0	2

Recap: Categorization

We have too many different **Ages**, we map them into buckets.

We wish to have 5-year buckets, how many buckets do we need?

```
array([ 0.,  5., 10., 15., 20., 25., 30., 35., 40., 45., 50., 55., 60.,
        65., 70., 75., 80.])
```

Recap: Categorization

Let's apply our categorization to the **Age** column values, by creating a new column **CatAge**:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	CatSex	CatEmbarked	CatAge
0	1	0	3	Braund, Mr. Owen Harris	male	22.000000	1	0	A/5 21171	7.2500	NaN	S	0	0	4
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.000000	1	0	PC 17599	71.2833	C85	C	1	1	7
2	3	1	3	Heikkinen, Miss. Laina	female	26.000000	0	0	STON/O2. 3101282	7.9250	NaN	S	1	0	5
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.000000	1	0	113803	53.1000	C123	S	1	0	6
4	5	0	3	Allen, Mr. William Henry	male	35.000000	0	0	373450	8.0500	NaN	S	0	0	6

Recap: Categorization

We have too many different **Fares**, we map them into buckets.

We wish to have 10-dollar buckets, how many buckets do we need?

```
array([  0.,  10.,  20.,  30.,  40.,  50.,  60.,  70.,  80.,  90., 100.,
       110., 120., 130., 140., 150., 160., 170., 180., 190., 200., 210.,
       220., 230., 240., 250., 260., 270., 280., 290., 300., 310., 320.,
       330., 340., 350., 360., 370., 380., 390., 400., 410., 420., 430.,
       440., 450., 460., 470., 480., 490., 500., 510., 520.])
```

Recap: Categorization

Let's apply our categorization to the **Fare** column values, by creating a new column **CatFare**:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	CatSex	CatEmbarked	CatAge	CatFare
0	1	0	3	Braund, Mr. Owen Harris	male	22.000000	1	0	A/5 21171	7.2500	NaN	S	0	0	4	0.0
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.000000	1	0	PC 17599	71.2833	C85	C	1	1	7	7.0
2	3	1	3	Heikkinen, Miss. Laina	female	26.000000	0	0	STON/O2. 3101282	7.9250	NaN	S	1	0	5	0.0
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.000000	1	0	113803	53.1000	C123	S	1	0	6	5.0

Recap: Visualizing Correlations

Let's remove all columns we no longer need from the old dataset: **14.2**

All of these columns have been categorized!

	Survived	Pclass	SibSp	Parch	CatSex	CatEmbarked	CatAge	CatFare
0	0	3	1	0	0	0	4	0.0
1	1	1	1	0	1	1	7	7.0
2	1	3	0	0	1	0	5	0.0
3	1	1	1	0	1	0	6	5.0
4	0	3	0	0	0	0	6	0.0
...	...	...	...	...	...	...	...	...
886	0	2	0	0	0	0	5	1.0
887	1	1	0	0	1	0	3	2.0
888	0	3	1	2	1	0	4	2.0
889	1	1	0	0	0	1	5	2.0
890	0	3	0	0	0	2	6	0.0

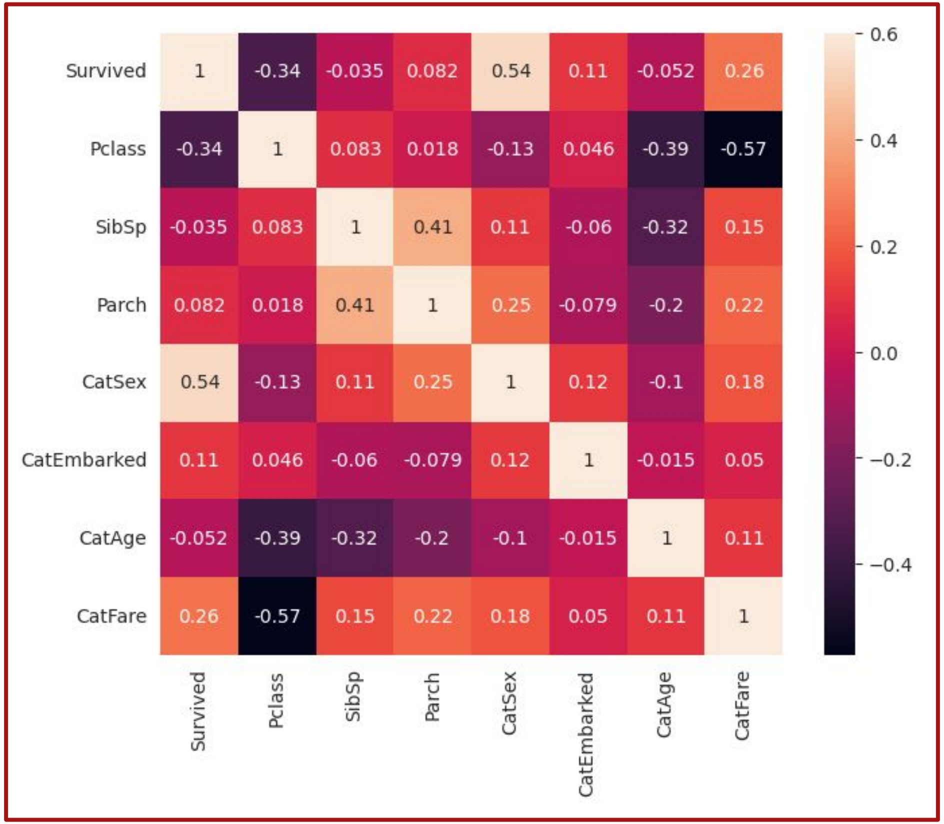
Recap: Visualizing Correlations

Understanding how **features relate to each other** is key in **prediction**.

We'll use **heatmaps** and custom plots to **visualize correlations** and interactions between variables.

Let's see how our categorized columns correlate to the **survival of passengers**: **14.3**

Visualizing Correlations



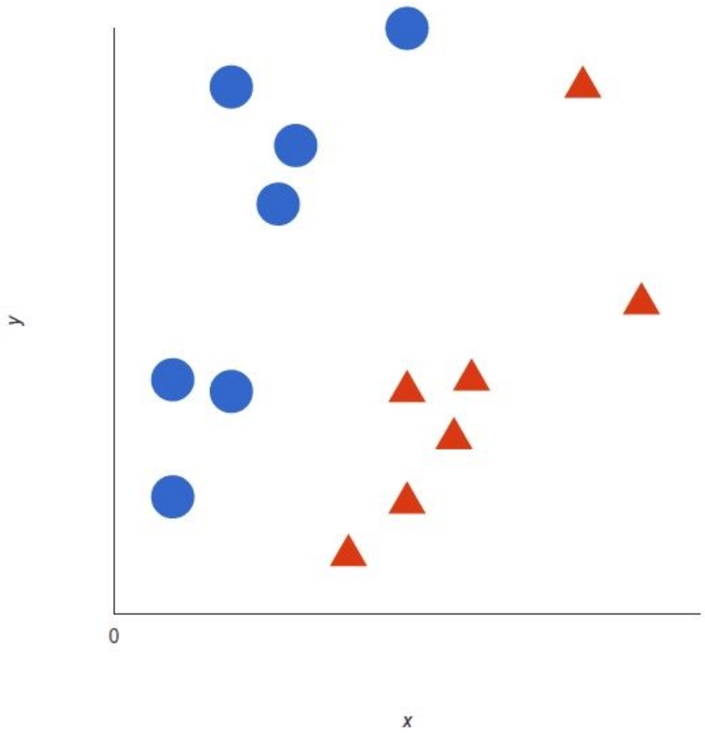
Classifiers

A classifier in machine learning is an algorithm that automatically orders or **categorizes data** into one or more of a set of "**classes**."

https://monkeylearn.com/blog/what-is-a-classifier/

Classifiers

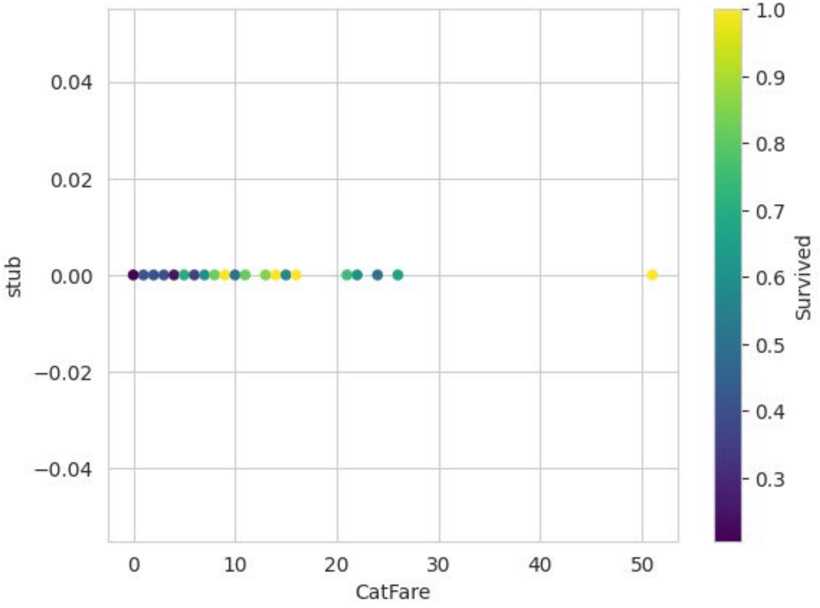
How would you think of an algorithm that classifies this data?



1-Dimensional Classification

Let's look at our dataset:

14.4



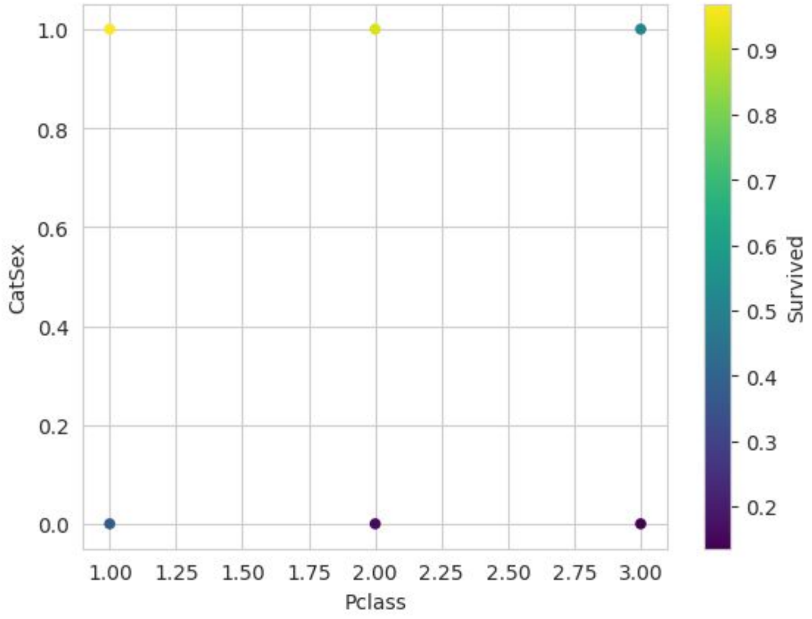
Explore **changing** the category: what do you see?

Are they all equally easy/hard?

2-Dimensional Classification

Let's look at our dataset:

14.5



Explore **changing** the category: what do you see?

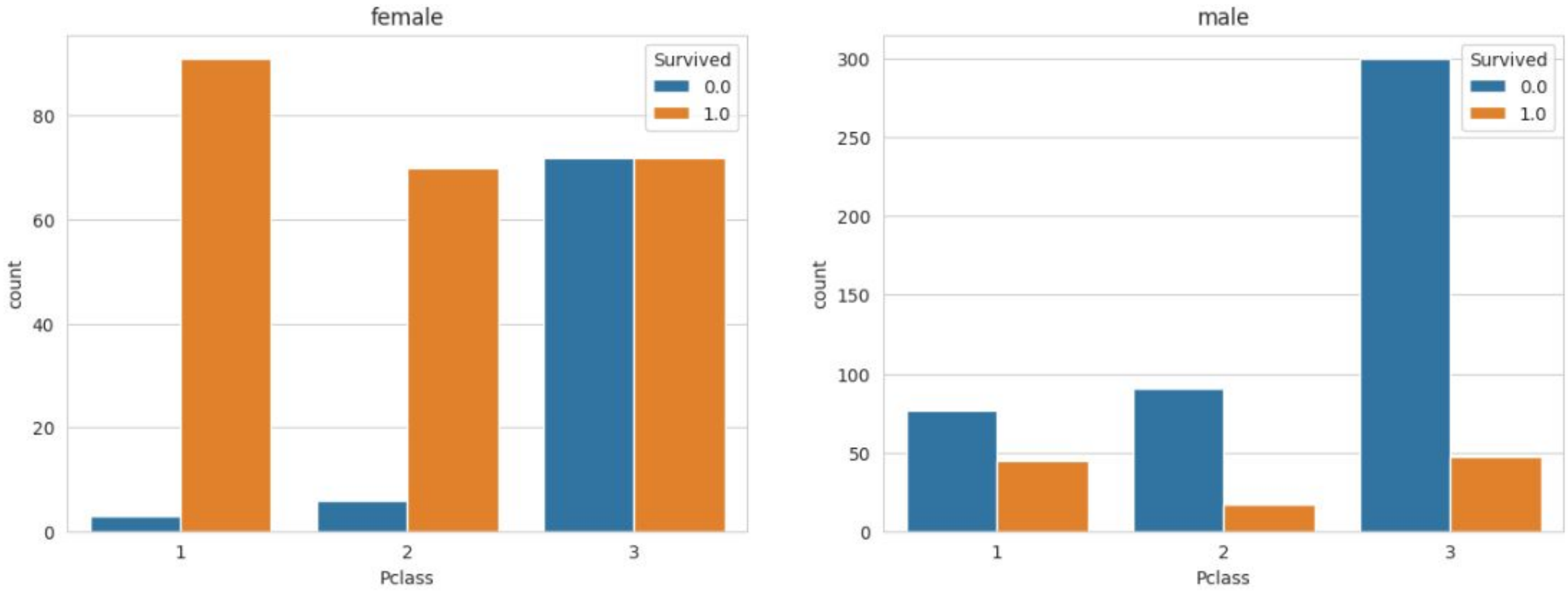
Then specifically chart:

```
plot_2d(x='Pclass', y='CatSex')
```

2-Dimensional Classification

We need to consider more dimensions to our data:

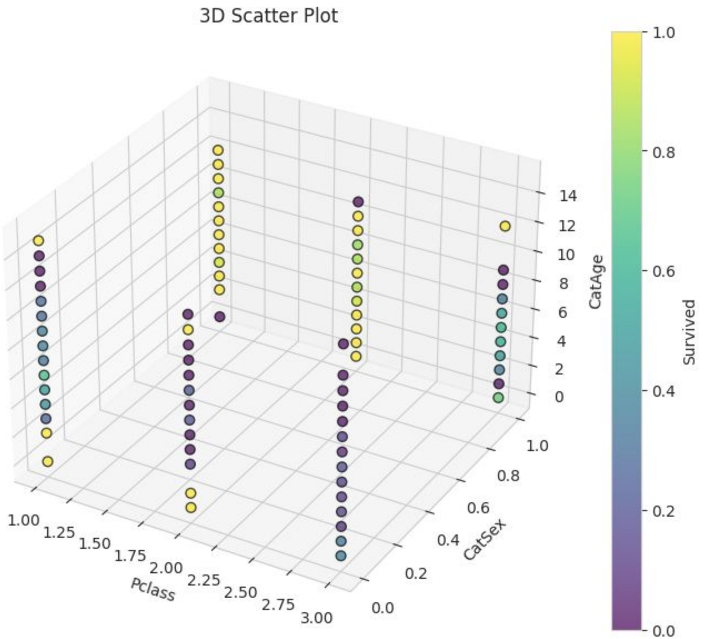
14.6



3-Dimensional Classification

Let’s look at our dataset:

14.7



Explore **changing** the category: what do you see?

Let’s understand how **machines** do it!

ML Classifiers

We must decide on which **features** we consider in the classification problem.

Then we must decide what we **classify against**.

14.8

```
Columns = ['Parch', 'Pclass', 'SibSp', 'CatSex', 'CatEmbarked', 'CatAge', 'CatFare']
Label = 'Survived'
```

ML Classifiers

When we classify we **split our data into training and test sets**.

Why? **14.9**

14.9.1

	Parch	Pclass	SibSp	CatSex	CatEmbarked	CatAge	CatFare
794	0	3	0	0	0	4	0.0
212	0	3	0	0	0	4	0.0
480	2	3	5	0	0	1	4.0
4	0	3	0	0	0	6	0.0
890	0	3	0	0	2	6	0.0
...	...	...	...	...	...	...	...
128	1	3	1	1	1	4	2.0
376	0	3	0	1	0	4	0.0
315	0	3	0	1	0	5	0.0
861	0	2	1	0	0	4	1.0
0	0	3	1	0	0	4	0.0

623 rows × 7 columns

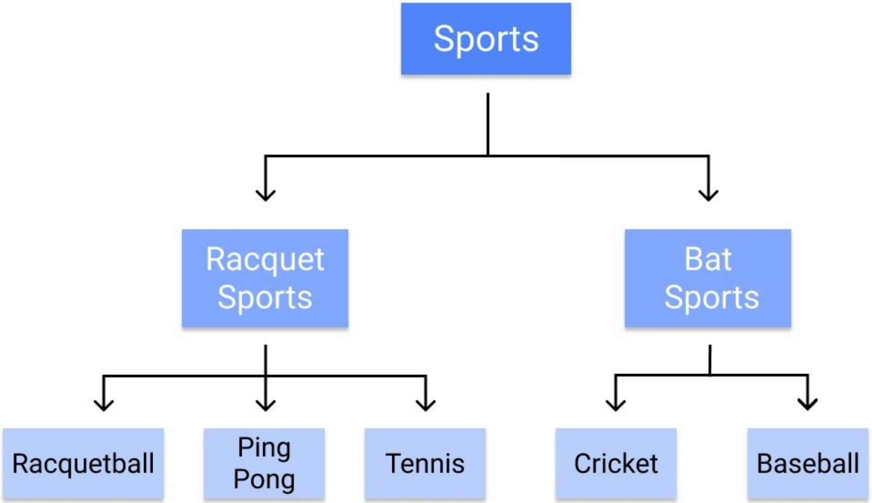
14.9.2

	Parch	Pclass	SibSp	CatSex	CatEmbarked	CatAge	CatFare
206	0	3	1	0	0	6	1.0
63	2	3	3	0	0	0	2.0
143	0	3	0	0	2	3	0.0
642	2	3	3	1	0	0	2.0
299	1	1	0	1	1	9	24.0
...	...	...	...	...	...	...	...
147	2	3	2	1	0	1	3.0
135	0	2	0	0	1	4	1.0
205	1	3	0	1	0	0	1.0
114	0	3	0	1	1	3	1.0
633	0	1	0	0	0	7	2.0

268 rows × 7 columns

Decision Tree Classifiers

It classifies data into **finer and finer categories**: from “tree trunk,” to “branches,” to “leaves.”

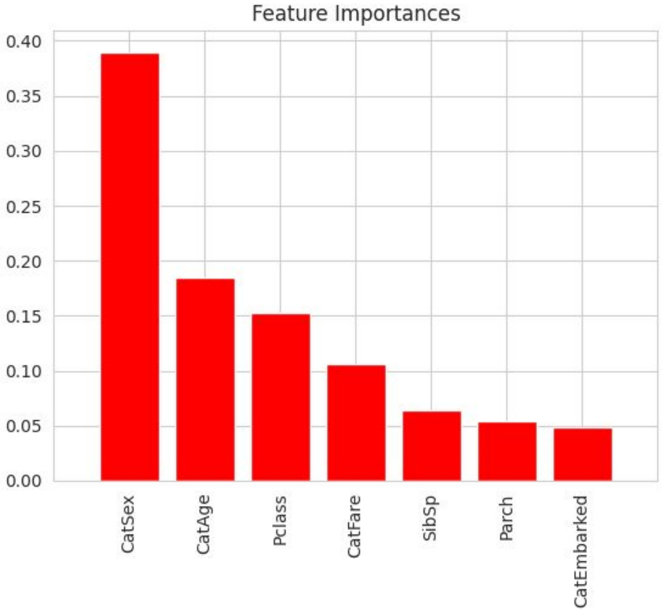


Decision Tree Classifiers

Create and fit the classifier: **14.10**

What features did it find most important?

14.10.1

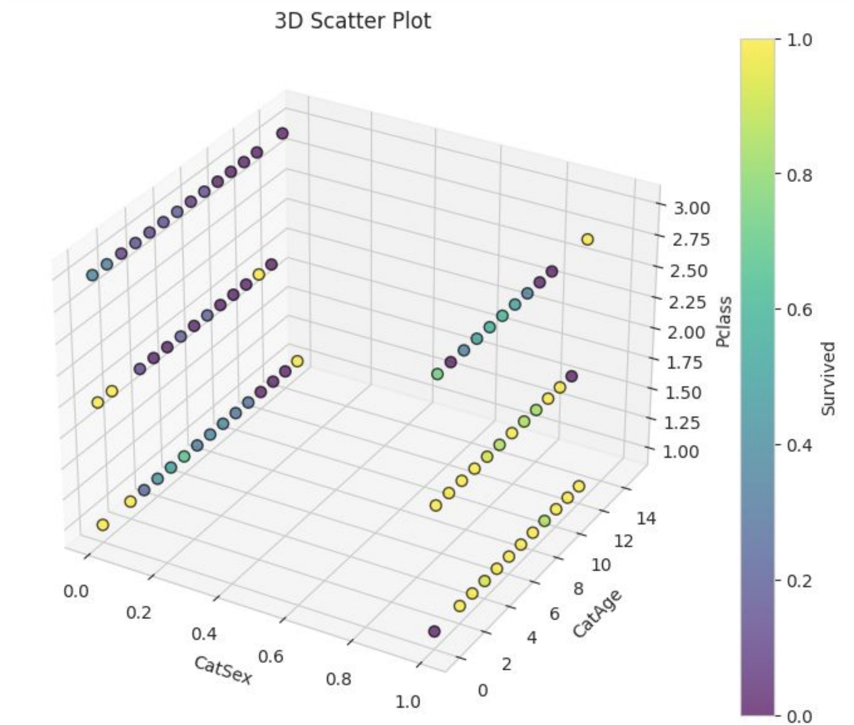


Exercise: Visualise them!

14.10.2

Decision Tree Classifiers

```
plot_3d('CatSex', 'CatAge', 'Pclass')
```



Decision Tree Classifiers

Let's see how **good our model** is: **14.10.3**

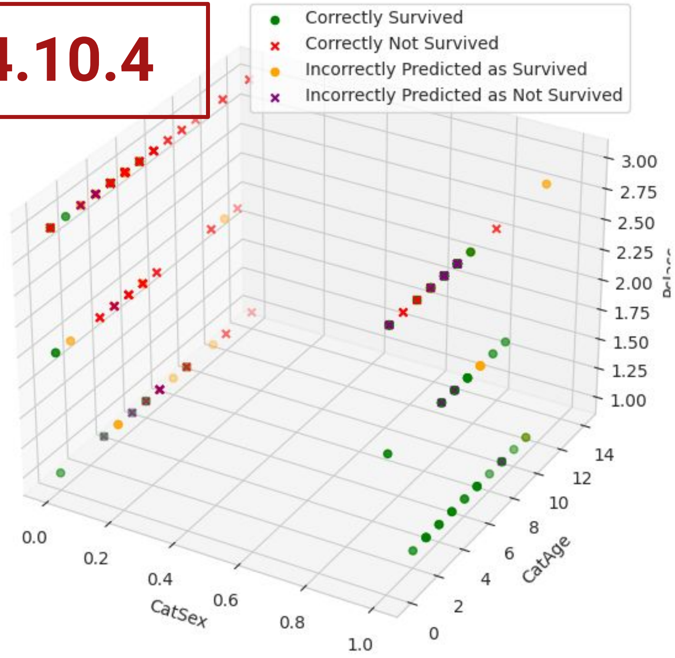
0.7835820895522388

What does this mean?

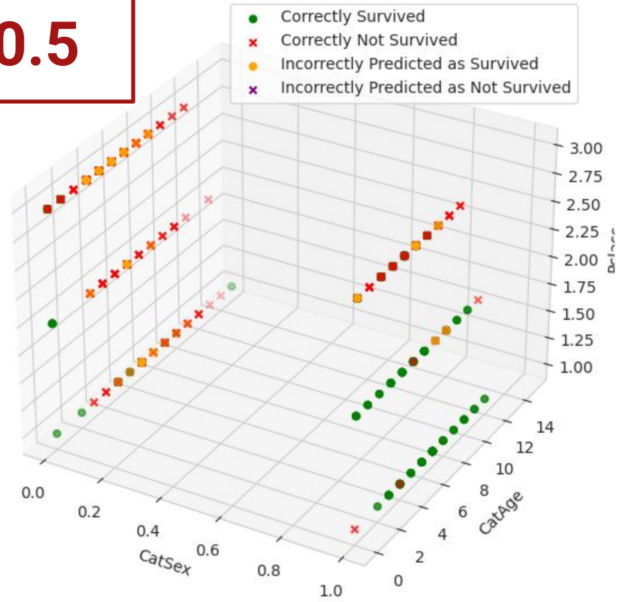
Decision Tree Classifiers

Let’s visualise our predictions:

14.10.4



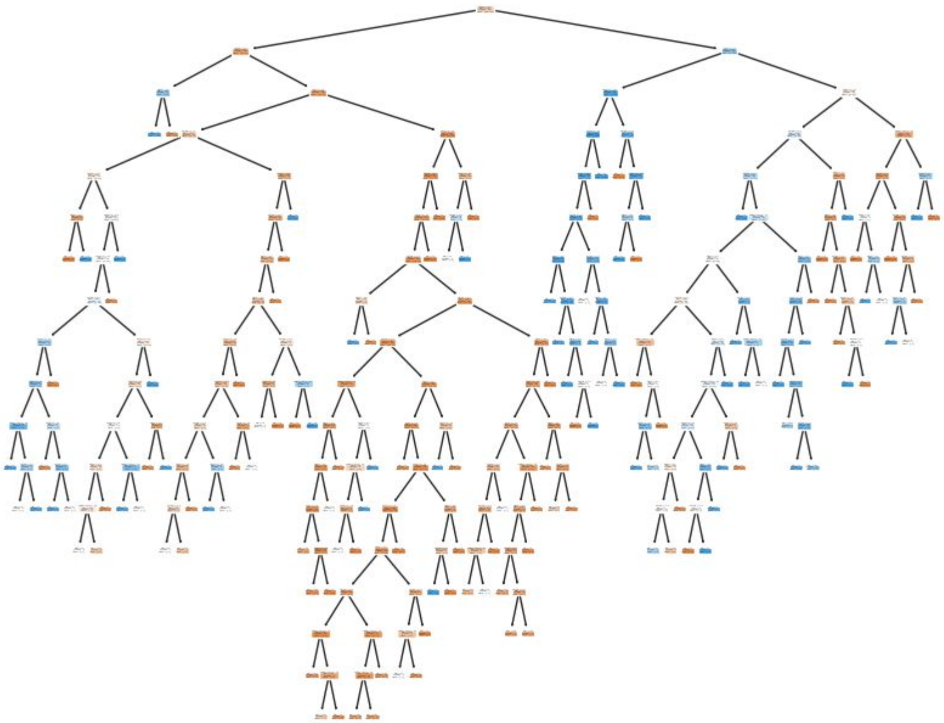
14.10.5



Decision Tree Classifiers

So what is our model doing? **14.10.6**

This visualises the full decision tree!



Random Forest

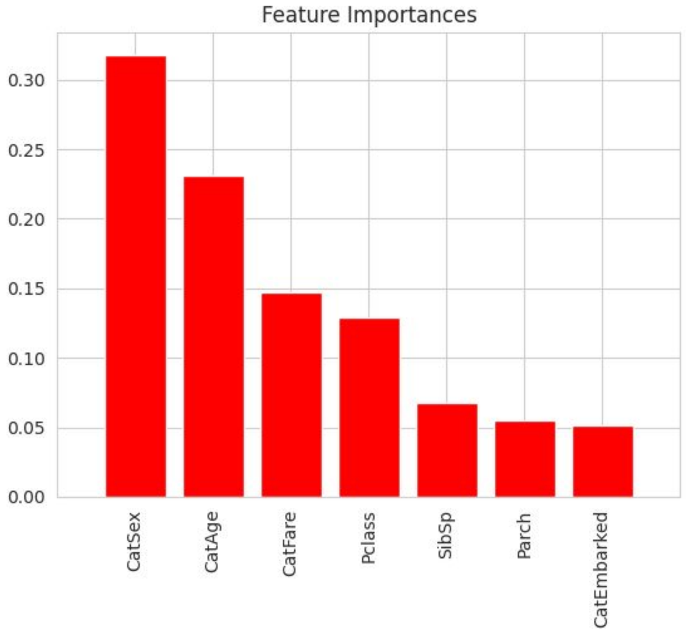
A **Random Forest** is like a **group decision-making** team in machine learning. It combines the opinions of many “trees” (individual models) to make **better predictions**, creating a more robust and accurate overall model.

Random Forest

Create and fit the classifier: **14.11**

What features did it find most important?

14.11.1

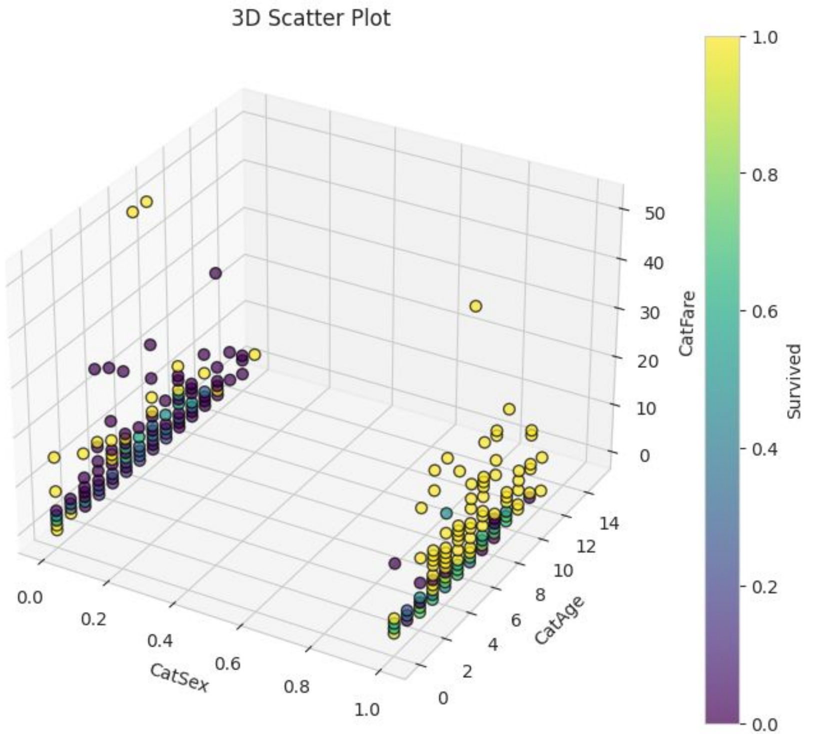


Exercise: Visualise them!

14.11.2

Random Forest

```
plot_3d('CatSex', 'CatAge', 'CatFare')
```



Random Forest

Let's see how **good our model** is: **14.11.3**

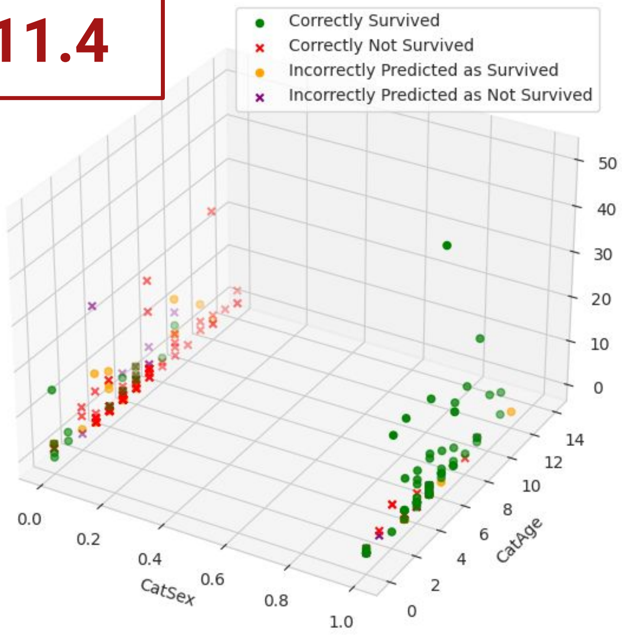
0.8059701492537313

What does this mean?

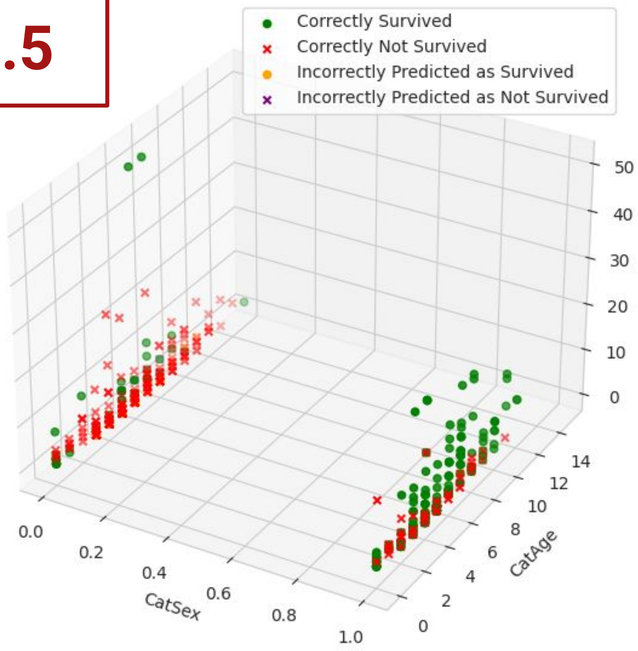
Random Forest

Let's visualise our predictions:

14.11.4

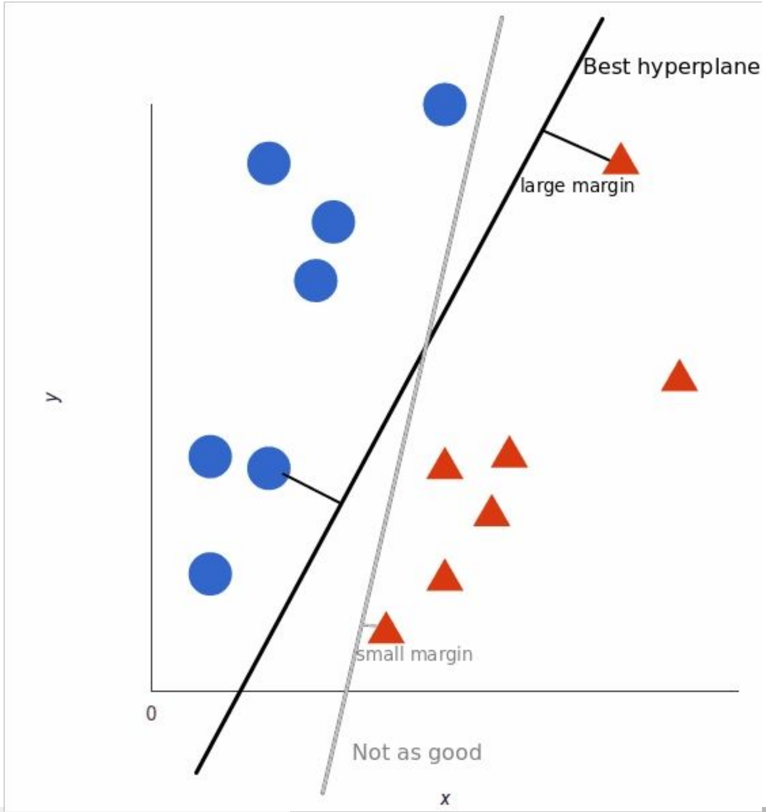


14.11.5



Support Vector Machines

SVM algorithms classify data and train models within super finite degrees of polarity, creating a **3-dimensional classification model** that goes beyond just X/Y predictive axes.



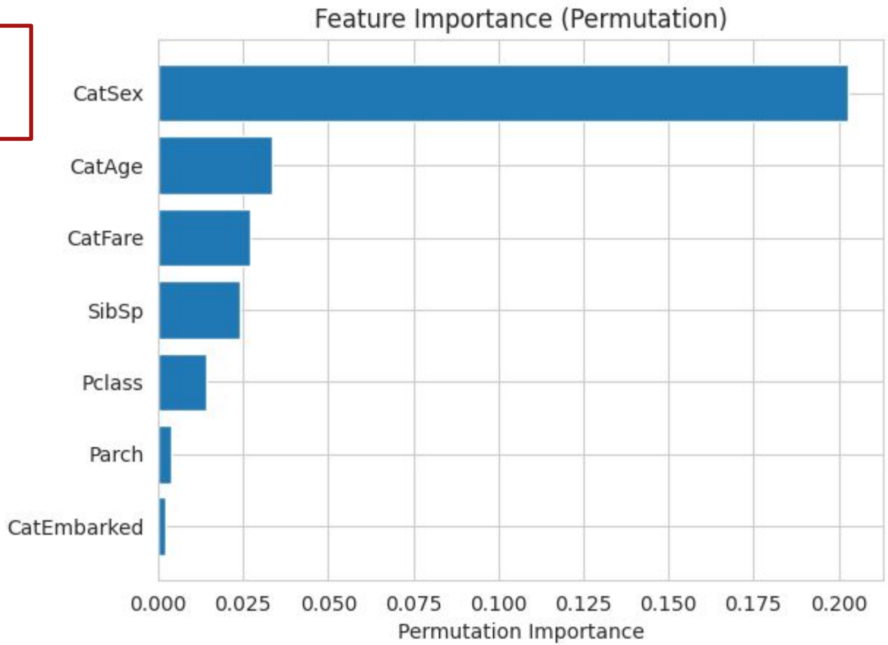
Support Vector Machines

Create and fit the classifier: **14.12**

What features did it find most important? **14.12.1**

Let's see how **good our model** is: **14.12.2**

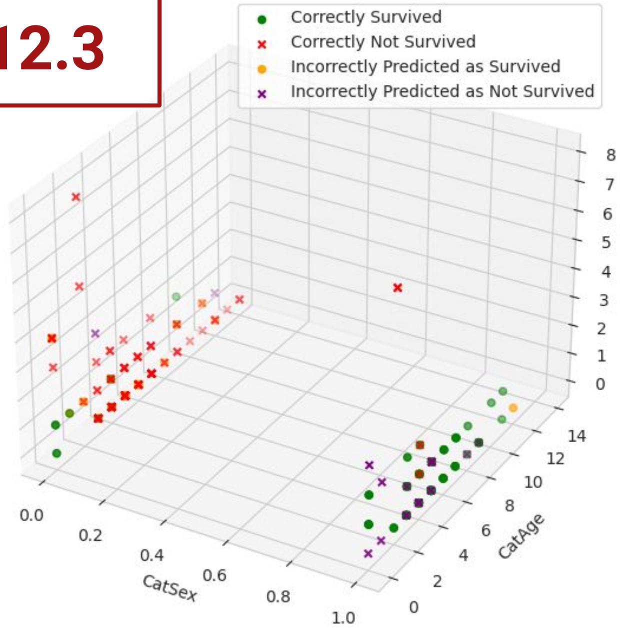
0.8059701492537313



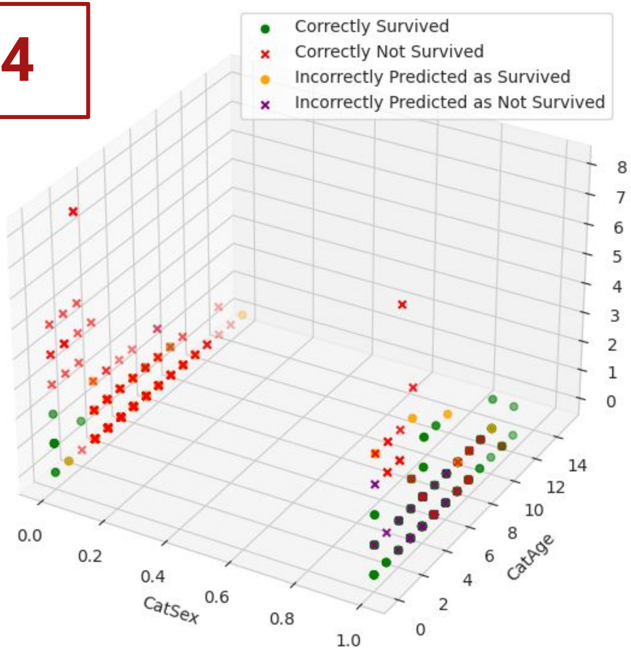
Support Vector Machines

Let’s visualise our predictions:

14.12.3



14.12.4



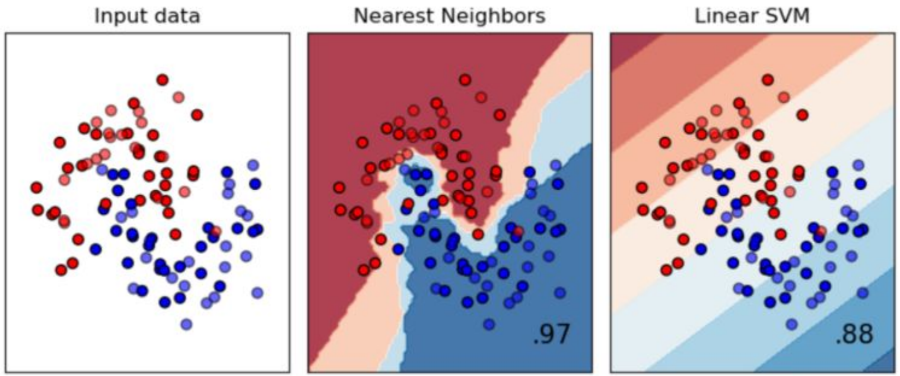
K-Nearest Neighbors

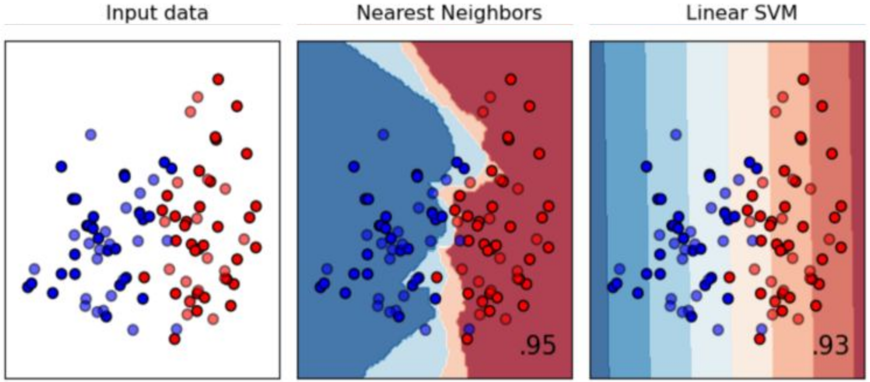
K-nearest neighbors (k-NN) is a pattern recognition algorithm that stores and learns from training data points by **calculating how they correspond to other data** in n-dimensional space. K-NN aims to find the **k closest related data points** in future, unseen data.

K-Nearest Neighbors

K-nearest neighbors (k-NN) is a pattern recognition algorithm that stores and learns from training data points by **calculating how they correspond to other data** in n-dimensional space. K-NN aims to find the **k closest related data points** in future, unseen data.

K-Nearest Neighbors





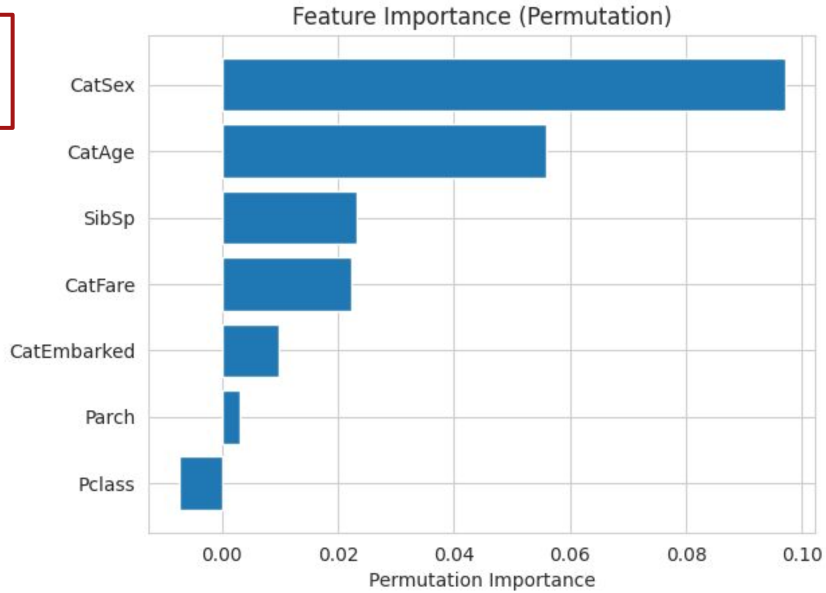
K-Nearest Neighbors

Create and fit the classifier: **14.13**

What features did it find most important? **14.13.1**

Let's see how **good our model** is: **14.13.2**

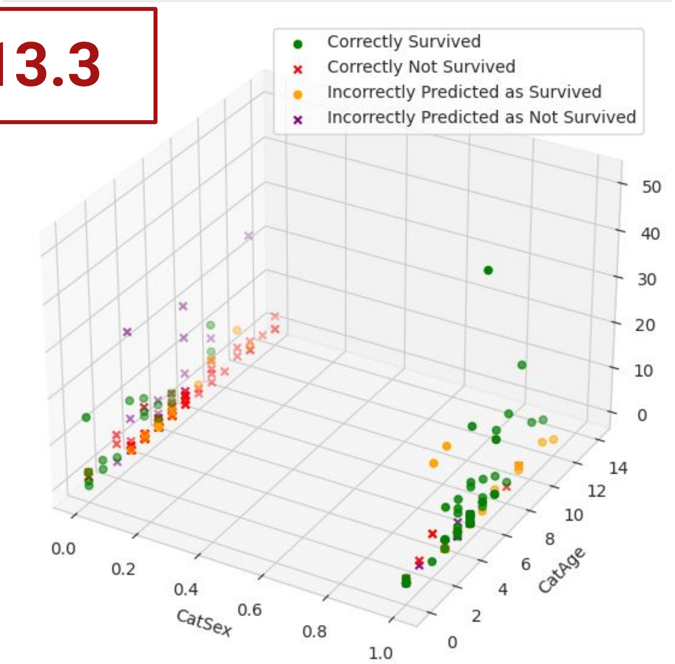
0.7350746268656716



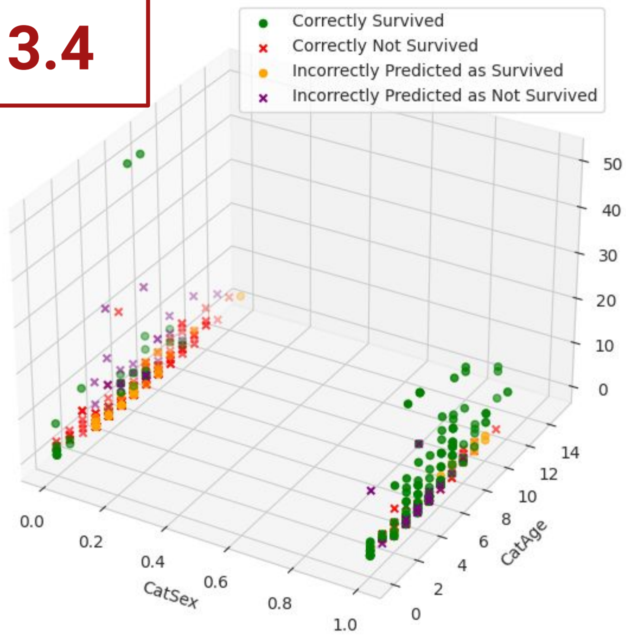
K-Nearest Neighbors

Let’s visualise our predictions:

14.13.3



14.13.4



End of Class

See you all next week!