COM4509/6509
Machine Learning and Adaptive Intelligence
Lecture 2b: End-to-End ML

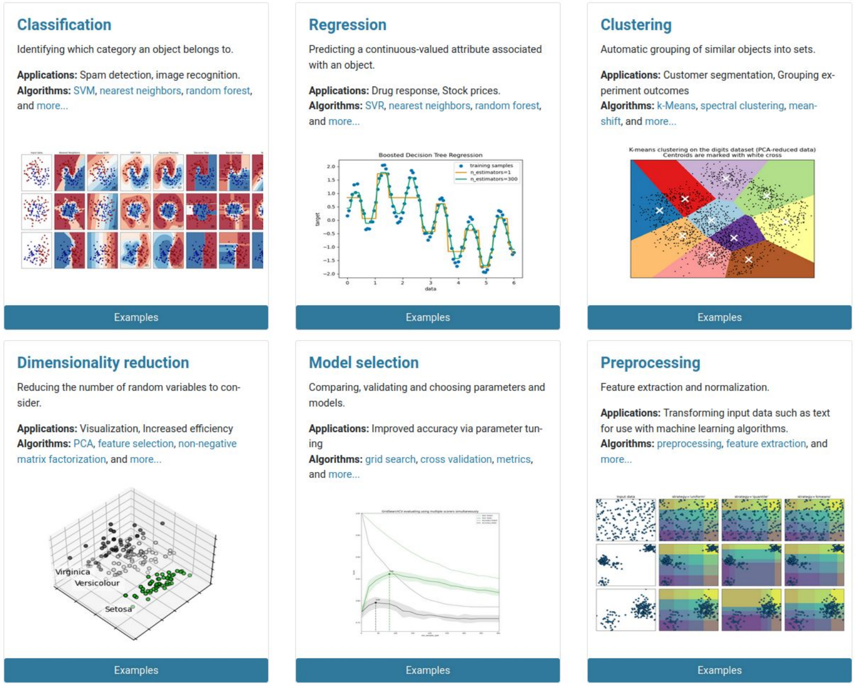
Mike Smith* and Matt Ellis

*m.t.smith@sheffield.ac.uk

Scikit-learn

We will use scikit-learn, a machine learning library for python.

- Supervised & unsupervised learning.
- Preprocessing and model selection.



Bike sharing demand prediction

Question:
How many people are going to rent a hire bike during the next hour, in Seoul?

Dataset:
There is a dataset of Seoul hire bike usage on the UCI ML repository.

Bike sharing demand prediction

Things we're missing:

1. Getting to see how the data is actually collected, and ideally visiting the city and seeing the system for ourselves.
2. Asking who wants to know, and why? How are they going to interpret it? What will be the consequences if it's wrong? What do they *really* want to know? Do we care about uncertainty?

What's the problem? _Too many/few bikes.

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality → Visualise, & explore

- over a day/year?
correlations
⋮

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

Visualise, & explore

• over a day/year?
correlations
⋮

unusable

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

• over a day/year?
correlations
⋮

Visualise, & explore

unusable

usable?

Prepare data

cleaning & feature engineering

Remove errors

eg discretise rainfall?

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

• over a day/year?
correlations
⋮

Visualise, & explore

unusable

usable?

Prepare data

cleaning & feature engineering

Remove errors

eg discretise rainfall?

Develop Model

+ You might want to compare to other models

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

Visualise, & explore

- over a day/year?
- correlations
- ⋮

usable?

unusable

Develop Model

+[You might want to compare to other models]

Prepare data

cleaning & feature engineering

Remove errors

eg discretise rainfall?

Train/select Model & Hyperparameters

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality

• over a day/year?
correlations
⋮

Visualise, & explore

unusable

usable?

Develop Model

+ [You might want to compare to other models]

Prepare data

cleaning & feature engineering

Remove errors

eg discretise rainfall?

Train/select Model & Hyperparameters

Test on <u>held out</u> test data

! DO NOT CHANGE MODEL BASED ON PERFORMANCE ON TEST DATA

What's the problem? — Too many/few bikes.

What do we want to know? — How many bikes will be used in next few hours?

Collect relevant data, with adequate quality → Visualise, & explore

- over a day/year?
- correlations
- ⋮

Visualise, & explore → Develop Model

\+ [You might want to compare to other models]

Visualise, & explore → usable? → Prepare data

Visualise, & explore → unusable → Collect relevant data, with adequate quality

Prepare data

- cleaning & feature engineering
- Remove errors
- eg discretise rainfall?

Prepare data → Train/select Model & Hyperparameters → Test on <u>held out</u> test data → Present/explain solution.

- Does it work?

Develop Model → Train/select Model & Hyperparameters

! DO NOT CHANGE MODEL BASED ON PERFORMANCE ON TEST DATA

What’s the problem?

Ask:

- What are current solutions… [to this or comparable problems?]
- How is performance measured?
- What is the minimum acceptable performance?

Also, issues from last time:

- Data shift (is bike hire data from 10 years ago relevant?)
- Are you interpolating or extrapolating? [e.g. a particularly rainy day]
- Probably still need a human in the loop, e.g. will the machine know about an unexpected bank-holiday? Or a transport strike? Etc.

Assessing Prediction Quality: Regression

- Example: I’m trying to predict how long it takes to cycle to work, so I can get to meetings not too early, but not too late.
- This is a regression problem (I’m trying to predict a continuous variable).
- I’ve two models I’m comparing:
    - a simple linear regression (inputs are time of day and rainfall)
    - a complex agent based model that simulates all the traffic in the city.
- I want to compare their prediction quality.

Assessing Prediction Quality: Regression

- I have some held out test data & the associated predictions for each model:

Date	Test Data / mins	Predictions / mins	
		Simple Model	Agent Based Model
05/09/22	14.1	14.7	16.2
06/09/22	14.3	14.6	16.1
07/09/22	14.6	15.3	12.4
08/09/22	14.8	15.3	15.4
09/09/22	14.7	15.2	16.2
12/09/22	15.4	16.2	16.2
13/09/22	15.6	16.1	15.3
14/09/22	15.6	14.1	16.2
15/09/22	55.2	15.4	16.1
16/09/22	15.3	15.3	14.1
19/09/22	16.4	16.3	[illegible]
20/09/22	16.3	[illegible]	[illegible]
21/09/22	17.1	[illegible]	[illegible]
22/09/22	[illegible]	[illegible]	[illegible]

Assessing Prediction Quality: Regression. RMSE

A very popular option to assess the prediction is to compute the **Root Mean Square Error**.

Prediction True value

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

Steps to compute the RMSE

1) Find the difference between your predictions and the true values (the error)
2) Square all these errors.
3) Find the average of these squared errors.
4) Square-root.

Assessing Prediction Quality: Regression. RMSE

- The RMSE is non-negative (we want it as small as possible).
- The standard deviation is the RMSE if your predictions are just the mean
  - So hopefully your RMSE is less than the standard deviation!

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

Assessing Prediction Quality: Regression. MAE

Another choice, that's more intuitive is the **mean absolute error**.

Prediction True value

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Steps to compute the MAE

1) Find the difference between your predictions and the true values (the error)
2) Find their absolute values (i.e. make the errors all positive)
3) Find the average of these absolute errors.

- The MAE is also non-negative (& we want it as small as possible too).

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\text{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Which you use depends on what you care about

- Are outliers a big problem? (use RMSE)
- To try to give an intuition:
  - True data: 3,4,5,6
  - Predictions: 4,3,4,5
  - Errors: 1,1,1,1

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Which you use depends on what you care about

- Are outliers a big problem? (use RMSE)
- To try to give an intuition:
    - True data: 3,4,5,6
    - Predictions: 4,3,4,5
    - Errors: 1,1,1,1

(ACTIVITY: What is the RMSE and MAE?)

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Which you use depends on what you care about

- Are outliers a big problem? (use RMSE)
- To try to give an intuition:
  - True data: 3,4,5,6
  - Predictions: 4,3,4,5
  - Errors: 1,1,1,1
  - RMSE = 1 & MAE = 1

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Which you use depends on what you care about

- Are outliers a big problem? (use RMSE)
- To try to give an intuition:
    - True data: 3,4,5,6
    - Predictions: 4,3,4,5
    - Errors: 1,1,1,1
    - RMSE = 1 & MAE = 1

- True data: 3,4,5,6
- Predictions: 3,4,5,2
- Errors: 0,0,0,4

(ACTIVITY: What is the RMSE and MAE?)

Assessing Prediction Quality: Regression. RMSE vs MAE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N}}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N}$$

Which you use depends on what you care about

- Are outliers a big problem? (use RMSE)
- To try to give an intuition:
  - True data: 3,4,5,6
  - Predictions: 4,3,4,5
  - Errors: 1,1,1,1
  - RMSE = 1 & MAE = 1

- True data: 3,4,5,6
- Predictions: 3,4,5,2
- Errors: 0,0,0,4
- RMSE = 2 & MAE = 1

In this example: The average error is 1, but the RMSE is 2. The RMSE penalises outliers more.

Assessing Prediction Quality: NLPD

- The **Negative Log Predictive Density** is a measure of error between a model's predictions and associated true values. Smaller values are better.
- Importantly: NLPD assesses the quality of the model's **uncertainty quantification**.
- It is used for both regression and classification.

$$\mathrm{NLPD} = -\sum_{i=1}^{N} \log p(y_i | \mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have a method (A) that classifies images as dogs or cats. Importantly it assigns **probabilities** to the two classes.

We show it a picture of three dogs and three cats.

Assessing Prediction Quality: NLPD

We have a method (A) that classifies images as dogs or cats. Importantly it assigns **probabilities** to the two classes.

We show it a picture of three dogs and three cats.

P(dog|image) = 0.9 0.4 0.7

Assessing Prediction Quality: NLPD

We have a method (A) that classifies images as dogs or cats. Importantly it assigns **probabilities** to the two classes.

We show it a picture of three dogs and three cats.

P(dog|image) = 0.9 0.4 0.7

P(cat|image) = 0.8 0.4 0.3

Assessing Prediction Quality: NLPD

We have a method (A) that classifies images as dogs or cats. Importantly it assigns **probabilities** to the two classes.

We show it a picture of three dogs and three cats.

P(dog|image) =	0.9	0.4	0.7			
P(cat|image) =				0.8	0.4	0.3

We're asking the model for the probability of the correct label.

Assessing Prediction Quality: NLPD

We have a method (A) that classifies images as dogs or cats. Importantly it assigns **probabilities** to the two classes.

We show it a picture of three dogs and three cats.

P(dog|image) = 0.9 0.4 0.7

P(cat|image) = 0.8 0.4 0.3

$$\mathrm{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Compute NLPD:
-(log(0.9)+log(0.4)+log(0.7)+log(0.8)+log(0.4)+log(0.3))=3.72

We're asking the model for the probability of the correct label.

$$\mathrm{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have another method (B) that we want to try. We show it the same pictures.

P(dog|image) = 0.95 0.98 0.02

$$\text{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have another method (B) that we want to try. We show it the same pictures.

P(dog|image) =	0.95	0.98	0.02			
P(cat|image) =				0.99	0.96	0.96

$$\text{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have another method (B) that we want to try. We show it the same pictures.

P(dog|image) =	0.95	0.98	0.02			
P(cat|image) =				0.99	0.96	0.96

Compute NLPD:
-(log(0.95)+log(0.98)+log(0.02)+log(0.99)+log(0.96)+log(0.96))=4.08

$$\text{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have another method (B) that we want to try. We show it the same pictures.

P(dog|image) =	0.95	0.98	0.02			
P(cat|image) =				0.99	0.96	0.96

Compute NLPD:
-(log(0.95)+log(0.98)+log(0.02)+log(0.99)+log(0.96)+log(0.96))=4.08

Even though the accuracy is higher, it scores worse on the NLPD. It's overconfident.

$$\mathrm{NLPD} = -\sum_{i=1}^{N} \log p(y_i|\mathbf{x_i})$$

Assessing Prediction Quality: NLPD

We have another method (B) that we want to try. We show it the same pictures.

P(dog|image) =	0.95	0.98	0.02			
P(cat|image) =				0.99	0.96	0.96

Compute NLPD:
-(log(0.95)+log(0.98)+log(0.02)+log(0.99)+log(0.96)+log(0.96))=4.08

Even though the accuracy is higher, it scores worse on the NLPD. It's overconfident. Note, just assigning 0.5 to all the classes does worse than the first classifier (i.e. being less confident is also bad).

Assessing Prediction Quality: Classification

Confusion matrix. Each row is the true class and each column the predicted class. In method 'A' from the last few slides, one of the three dogs was classified correctly and two of the three cats.

		Predicted Class	
		Dog	Cat
True Class	Dog		
	Cat		

ACTIVITY! Fill this in!
Need a number in each cell

Assessing Prediction Quality: Classification

Confusion matrix. Each row is the true class and each column the predicted class. In method 'A' from the last few slides, one of the three dogs was classified correctly and two of the three cats.

Method A

True Class		Predicted Class: Dog	Predicted Class: Cat
	Dog	1	2
	Cat	1	2

Assessing Prediction Quality: Classification

Confusion matrix. Each row is the true class and each column the predicted class. In method 'A' from the last few slides, one of the three dogs was classified correctly and two of the three cats.

Method A

True Class		Predicted Class: Dog	Predicted Class: Cat
	Dog	1	2
	Cat	1	2

Method B

True Class		Predicted Class: Dog	Predicted Class: Cat
	Dog	2	1
	Cat	0	3

Assessing Prediction Quality: Classification

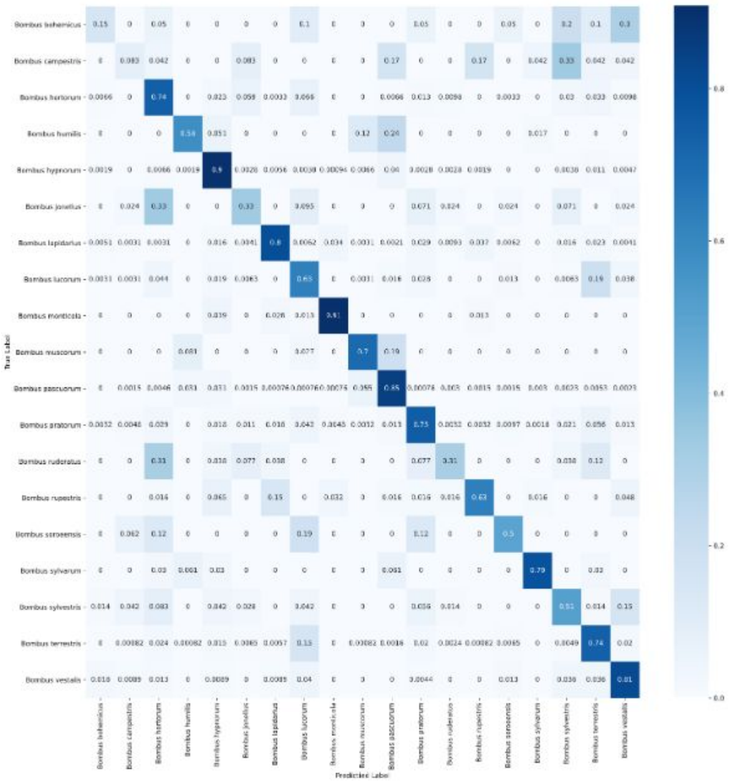
Confusion matrix. Each row is the true class and each column the predicted class. In method 'A' from the last few slides, one of the three dogs was classified correctly and two of the three cats.

The diagonal cells are the ones where the model has classified the labels correctly.

Assessing Prediction Quality: Classification

In multiclass situations it can help us see where the model is getting confused.

Table is from a student project classifying photos of bees.

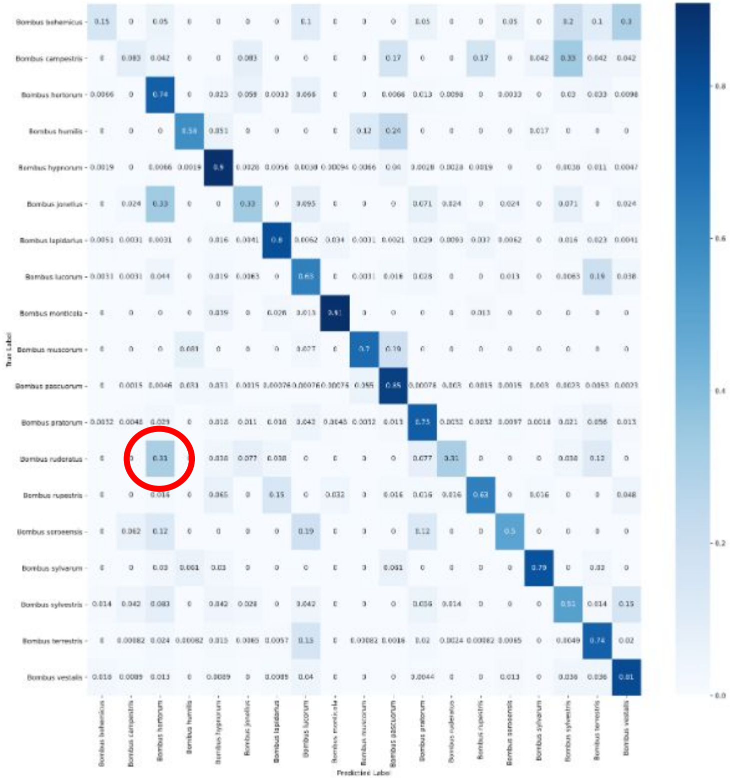


From Jennifer Ollett's dissertation (2021)

Assessing Prediction Quality: Classification

In multiclass situations it can help us see where the model is getting confused.

Table is from a student project classifying photos of bees.

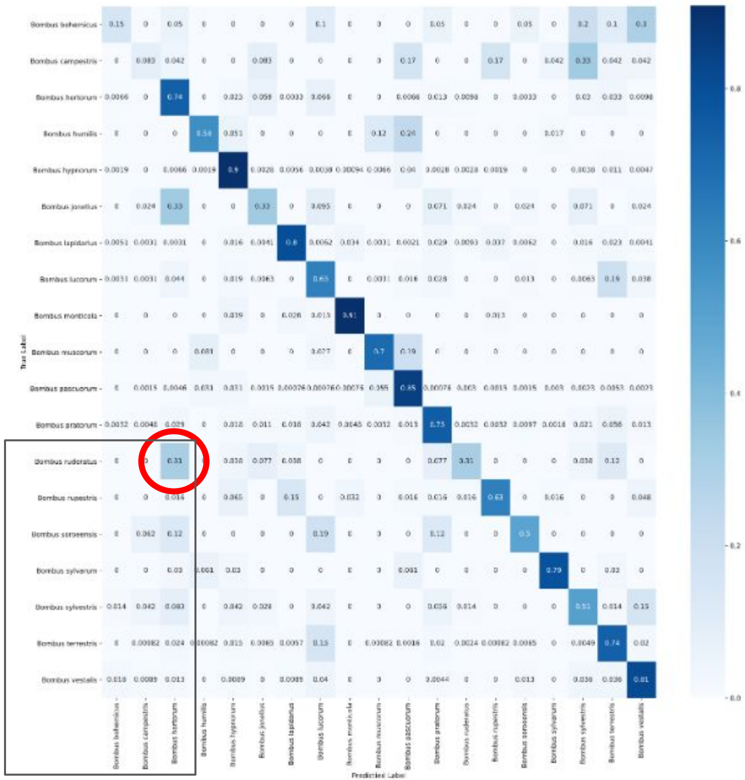


From Jennifer Ollett’s dissertation (2021)

Assessing Prediction Quality: Classification

In multiclass situations it can help us see where the model is getting confused.

Table is from a student project classifying photos of bees.



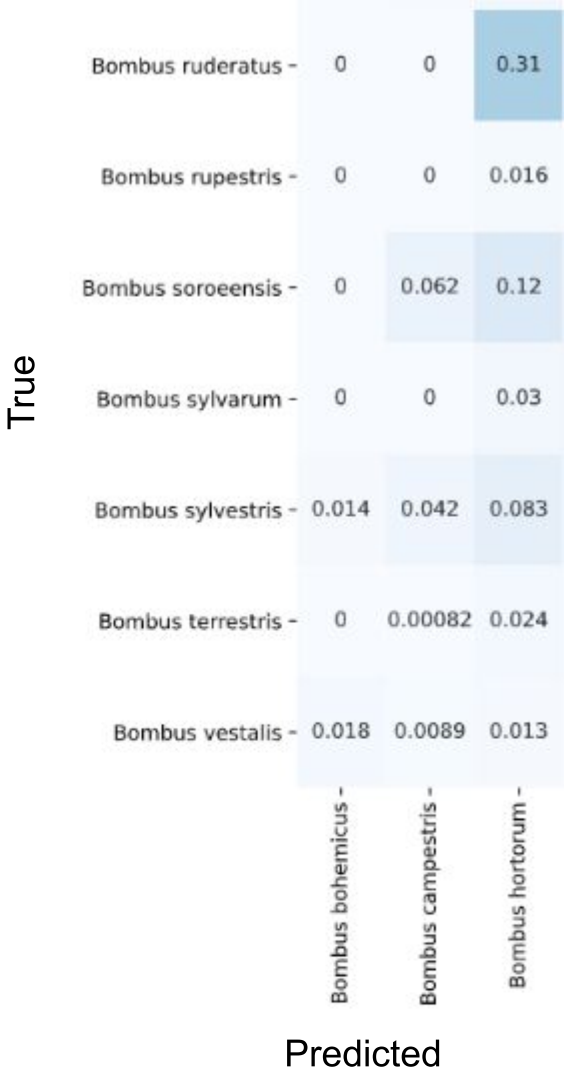
From Jennifer Ollett's dissertation (2021)

Assessing Prediction Quality

In multiclass situations it can help us see where the model is getting confused.

Table is from a student project classifying photos of bees.

Bombus ruderatus vs Bombus hortorum
(Ruderal bumblebee vs Garden Bumblebee)



True \ Predicted	Bombus bohemicus	Bombus campestris	Bombus hortorum
Bombus ruderatus	0	0	0.31
Bombus rupestris	0	0	0.016
Bombus soroeensis	0	0.062	0.12
Bombus sylvarum	0	0	0.03
Bombus sylvestris	0.014	0.042	0.083
Bombus terrestris	0	0.00082	0.024
Bombus vestalis	0.018	0.0089	0.013

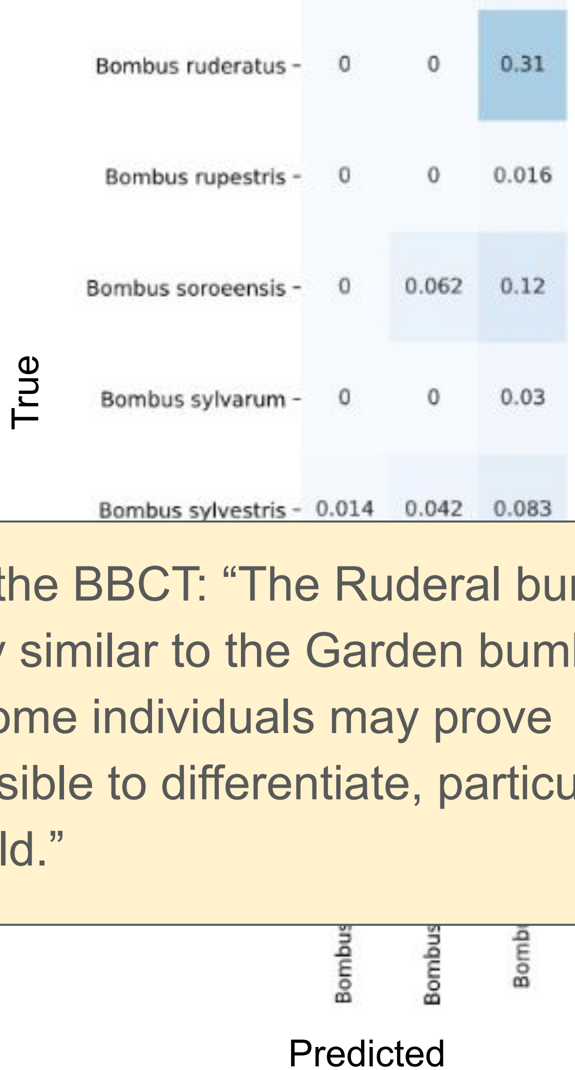
From Jennifer Ollett's dissertation (2021)

Assessing Prediction Quality

In multiclass situations it can help us see where the model is getting confused.

Table is from a student project classifying photos of bees.

Bombus ruderatus vs Bombus hortorum
(Ruderal bumblebee vs Garden Bumblebee)



From the BBCT: “The Ruderal bumblebee is very similar to the Garden bumblebee, and some individuals may prove impossible to differentiate, particularly in the field.”

dissertation
(2021)

Assessing Prediction Quality: Classification

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

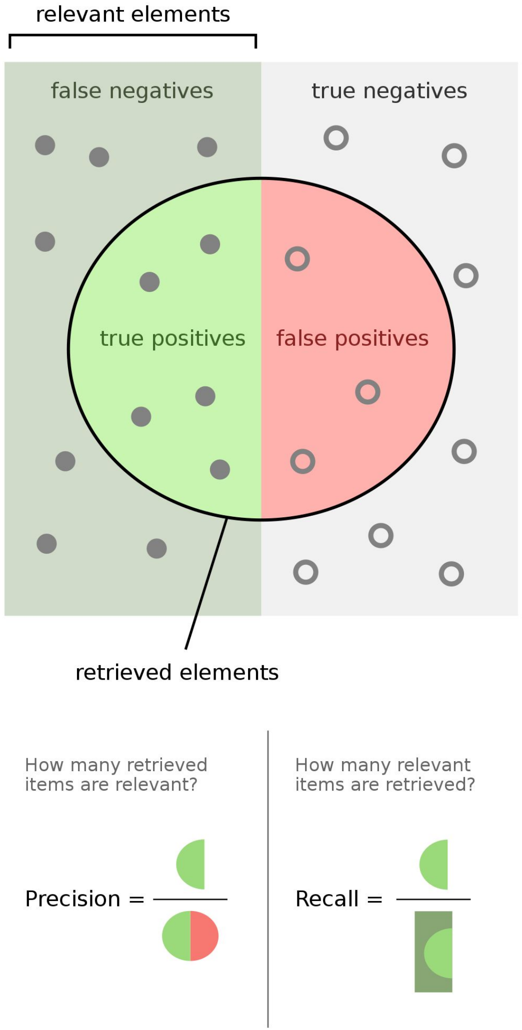
Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN)



Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

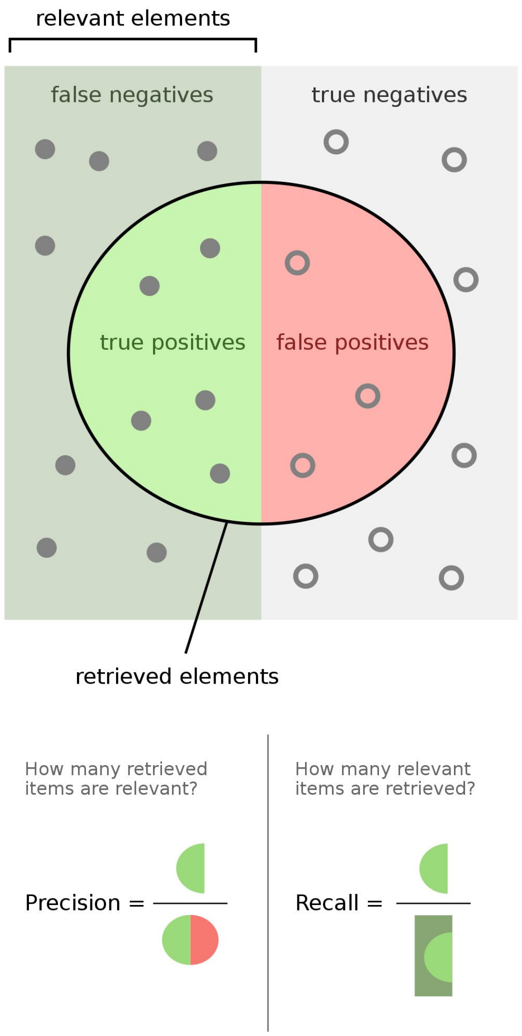
Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

Example

Spam filtering (True = spam).

- Do we want high precision/low recall?
- Or low precision/high recall?

ACTIVITY: Decide!



Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

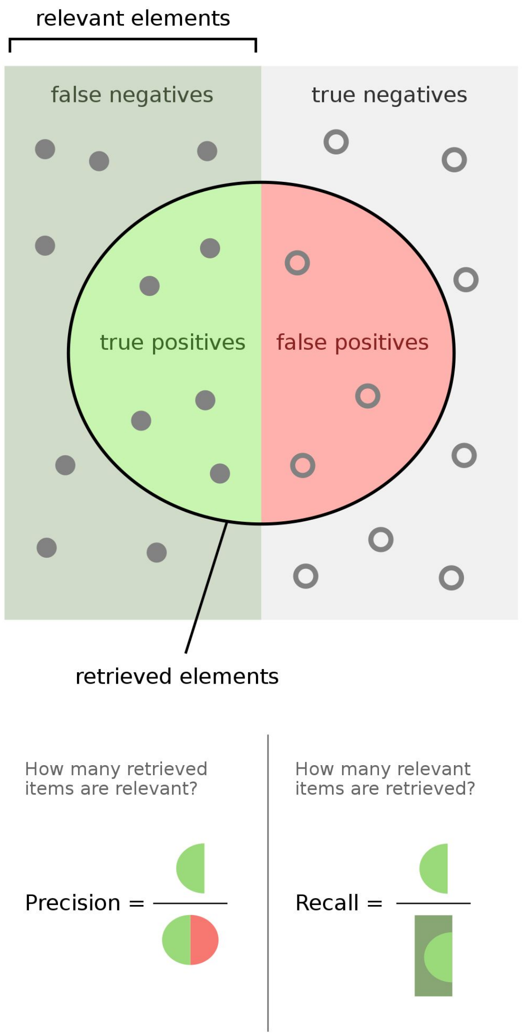
Example

Spam filtering (True = spam).

- Do we want high precision/low recall?
- Or low precision/high recall?

If we have low precision: Correctly labelled spam is a small proportion of items labelled as spam.

Bad



Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

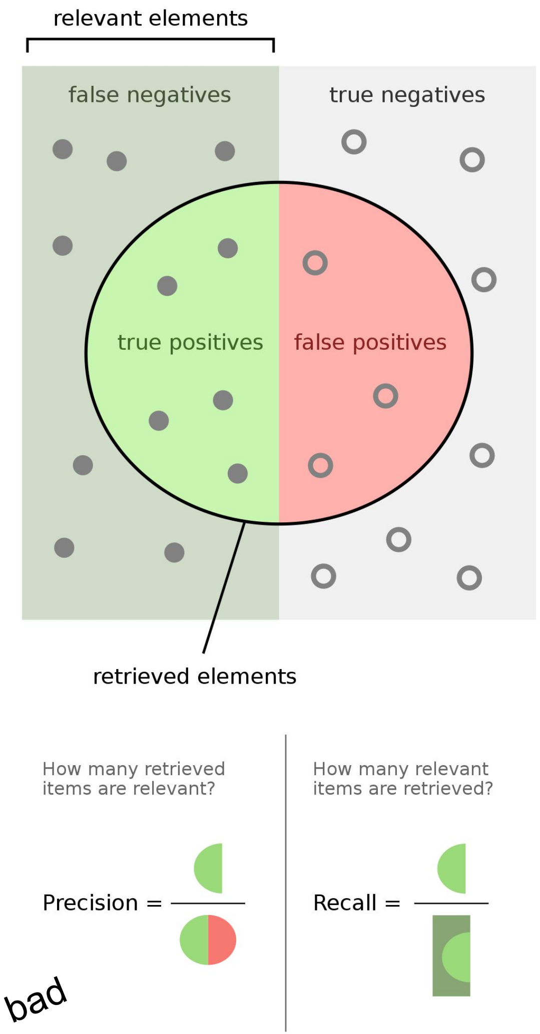
Example

Spam filtering (True = spam).

- Do we want high precision/low recall?
- Or low precision/high recall?

If we have low recall: Correctly labelled spam is a small proportion of all the real spam (so some spam gets into inbox)

Less bad



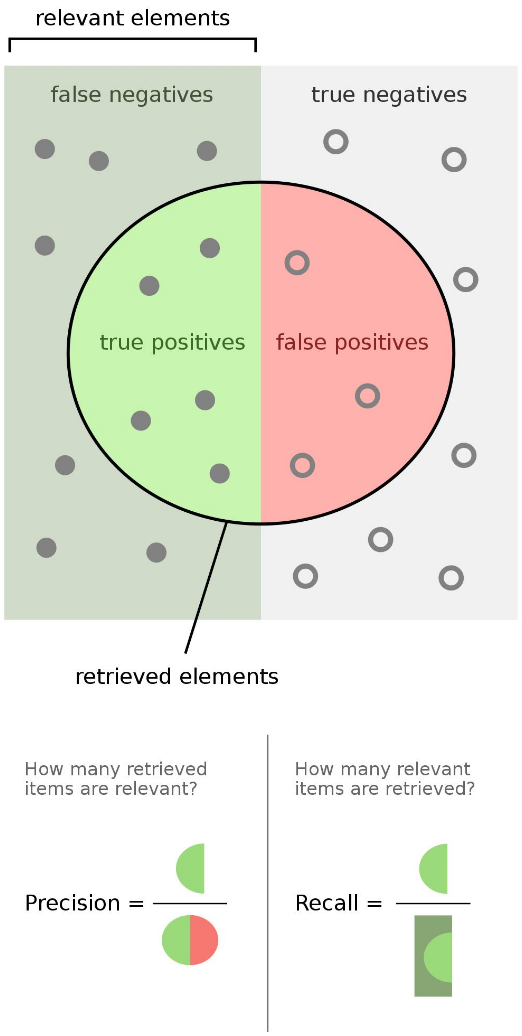
Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

Accuracy = TP + TN / (TP + TN + FP + FN)



Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN)

Accuracy = TP + TN / (TP + TN + FP + FN)

- Accuracy is a useful metric when errors predicting all classes are **equally important.**
- In the spam example, FP are worse that FN.
- Accuracy can be misleading with **imbalanced data**, e.g. you can have a high TP value and a low TN value, and your accuracy could still be high

Assessing Prediction Quality

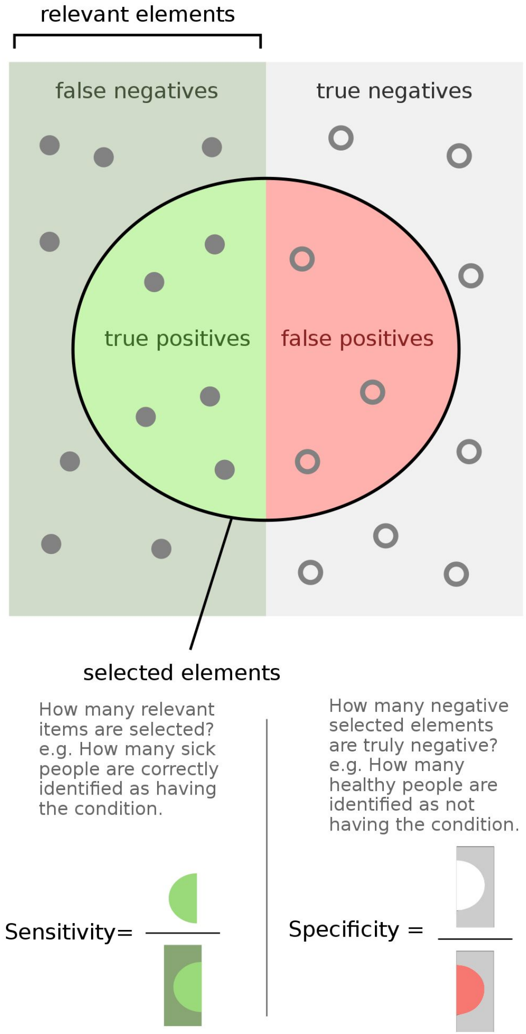
If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

Precision = TP / (TP + FP)
Recall = TP / (TP + FN) = Sensitivity

Accuracy = TP + TN / (TP + TN + FP + FN)

Specificity = TN / (TN + FP)



Assessing Prediction Quality

If your labels are true / false. Then you can think about:

- True positive (rate)
- False positive (rate)
- True negative (rate)
- False negative (rate)

If we define ‘positive’ as having the disease…

Precision = TP / (TP + FP)
- Out of those classified as having the disease how many really do.

Recall = TP / (TP + FN) = Sensitivity
- Probability of a positive test, given they have the disease

Specificity = TN / (TN + FP)
- Probability of a negative test, given they are well.

Example

We have a system that classifies mosquitos from their sounds.

Positive classification = Aedes aegypti (carries yellow fever).

Negative classification = anything else.

Of the 20 positive examples, 6 are correctly classified.

Of the 80 negative examples, 70 are correctly classified.

ACTIVITY

1) Write down the confusion matrix
2) What is the Accuracy?
3) What is the Precision?
4) What is the Recall / Sensitivity?
5) What is the Specificity?

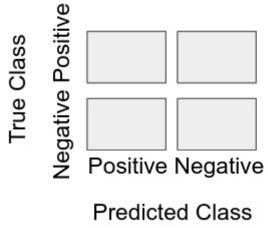
Reminder:

Precision = TP / (TP + FP)

Recall & Sensitivity = TP / (TP + FN)

Specificity = TN / (TN + FP)

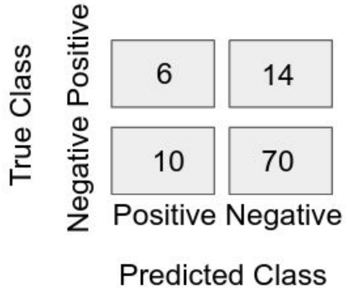
Accuracy = TP + TN / (TP + TN + FP + FN)



Vasconcelos, Dinarte, et al. "Locomobis: a low-cost acoustic-based sensing system to monitor and classify mosquitoes." *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2019.

Example

Of the 20 positive examples, 6 are correctly classified.
Of the 80 negative examples, 70 are correctly classified.



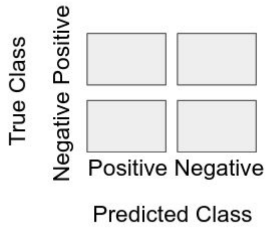
Accuracy = 76/100 = 76%
Precision = 6/16 = 38%
Recall and Sensitivity = 6/20 = 30%
Specificity = 70 / 80 = 88%

ACTIVITY

1) Write down the confusion matrix
2) What is the Accuracy?
3) What is the Precision?
4) What is the Recall / Sensitivity?
5) What is the Specificity?

Reminder:

Precision = TP / (TP + FP)
Recall & Sensitivity = TP / (TP + FN)
Specificity = TN / (TN + FP)
Accuracy = TP + TN / (TP + TN + FP + FN)



Vasconcelos, Dinarte, et al. "Locomobis: a low-cost acoustic-based sensing system to monitor and classify mosquitoes." *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2019.

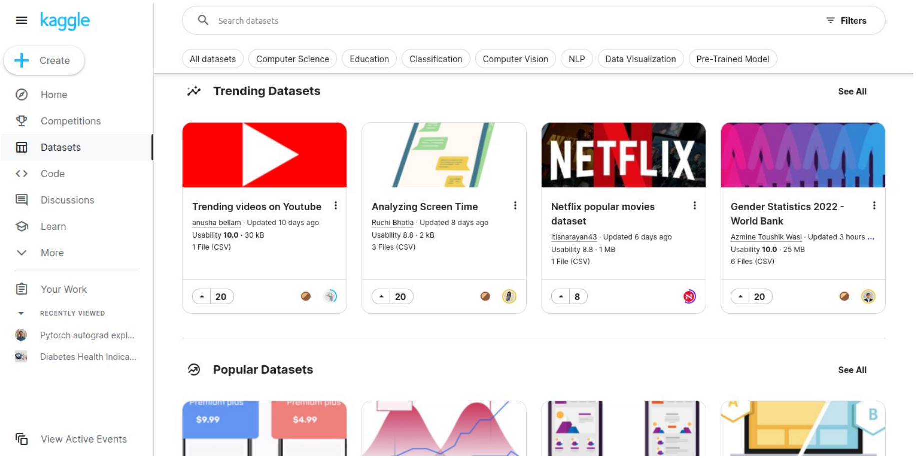
Getting the Data

- Quality? Quantity?
- Legal obligations [medical? GDPR? NDAs?]
- Anonymised?
- Remove a test set for later (leave it aside and come back to it at the very end)
- For generalisation - is there similar data elsewhere you could test on?

Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)



Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)

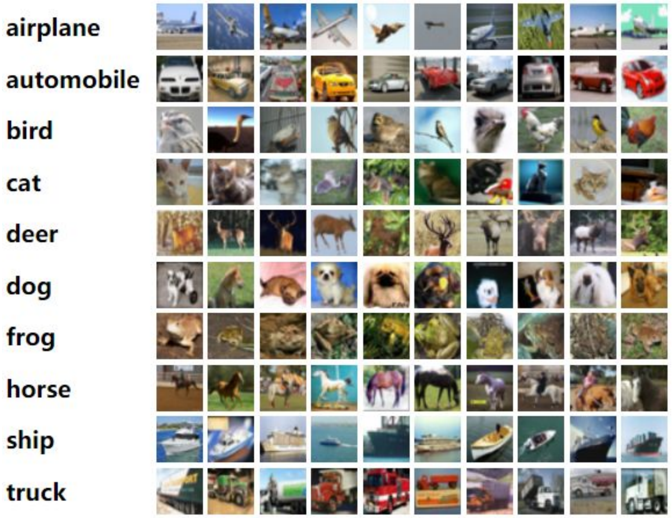
Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)
- Wikipedia has a list of datasets.

Famous datasets:

- CIFAR-10 (60,000 32x32 images. 10 classes)



Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)
- Wikipedia has a list of datasets.

Famous datasets:

- CIFAR-10 (60,000 32x32 images. 10 classes)
- ImageNet (14M big colour images. 20k classes)

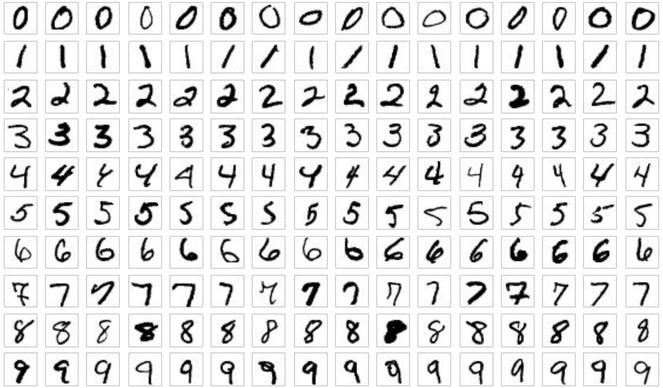
Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)
- Wikipedia has a list of datasets.

Famous datasets:

- CIFAR-10 (60,000 32x32 images. 10 classes)
- ImageNet (14M big colour images. 20k classes)
- MNIST (Handwritten digits 28x28, 10 classes)



Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)
- Wikipedia has a list of datasets.

Famous datasets:

- CIFAR-10 (60,000 32x32 images. 10 classes)
- ImageNet (14M big colour images. 20k classes)
- MNIST (Handwritten digits 28x28, 10 classes)
- Amazon reviews (link) (233M reviews)

Where to get data

If you want to play, there are fun datasets on:

- Kaggle (www.kaggle.com)
- UCI Machine Learning Repository (archive.ics.uci.edu/ml)
- Wikipedia has a list of datasets.

Famous datasets:

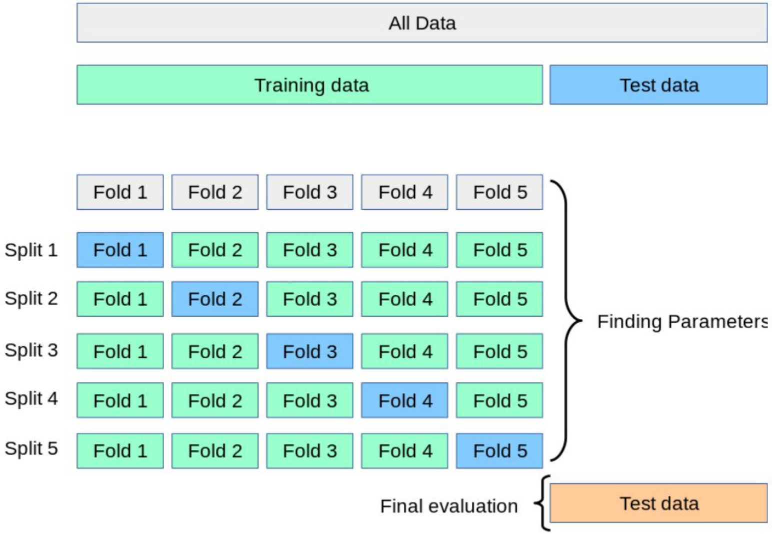
- CIFAR-10 (60,000 32x32 images. 10 classes)
- ImageNet (14M big colour images. 20k classes)
- MNIST (Handwritten digits 28x28, 10 classes)
- Amazon reviews (link) (233M reviews)
- MovieLens (link) (25M movie ratings: 62k movies, 162k users) …

	userId	movieId	rating	timestamp
0	1	1	4.0	964982703
1	1	3	4.0	964981247
2	1	6	4.0	964982224
3	1	47	5.0	964983815
4	1	50	5.0	964982931
5	1	70	3.0	964982400
6	1	101	5.0	964980868
7	1	110	4.0	964982176
8	1	151	5.0	964984041
9	1	157	5.0	964984100

codespeedy.com

Train / Validation / Test

- We’ve already discussed these datasets.
- You might split the training and validation with cross-validation.



Credit: From Mauricio’s slide.

Explore the Data

Look at each feature

- Type? (categorical, continuous, etc)
- Missing values?
- Types of noise / error (outliers?)
- Is it a useful feature?

Visualise

Returning to Sathishkumar et al. (2020)...

Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data

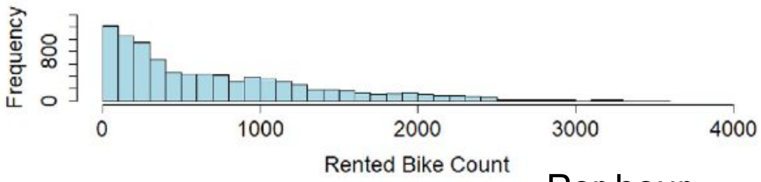
Returning to Sathishkumar et al. (2020)...

Table 1
Data variables and description.

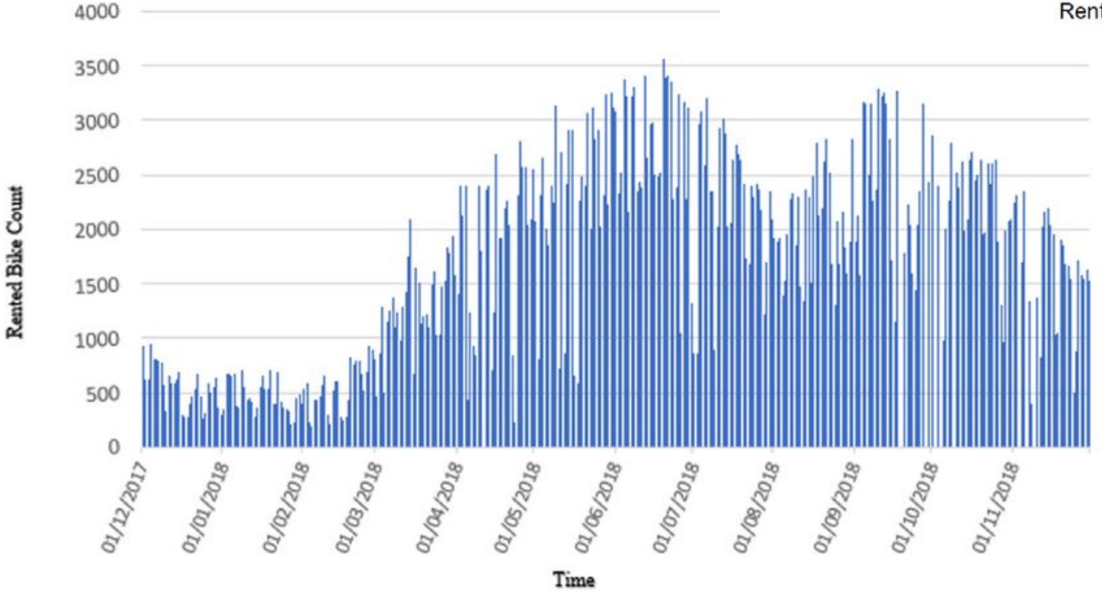
Parameters/Features	Abbreviation	Type	Measurement
Date	Date	year-month-day	–
Rented Bike count	Count	Continuous	0, 1, 2, ..., 3556
Hour	Hour	Continuous	0, 1, 2, ..., 23
Temperature	Temp	Continuous	°C
Humidity	Hum	Continuous	%
Windspeed	Wind	Continuous	m/s
Visibility	Visb	Continuous	10 m
Dew point temperature	Dew	Continuous	°C
Solar radiation	Solar	Continuous	MJ/m2
Rainfall	Rain	Continuous	Mm
Snowfall	Snow	Continuous	cm
Seasons	Seasons	Categorical	Autumn, Spring, Summer, Winter
Holiday	Holiday	Categorical	Holiday, Workday
Functional Day	Fday	Categorical	NoFunc, Func
Week status	Wstatus	Categorical	Weekday (Wday), Weekend (Wend)
Day of the week	Dweek	Categorical	Sunday, Monday, ..., Saturday

Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data



Per hour



Per hour

Fig. 2. Rented bike count measurement for the whole period.

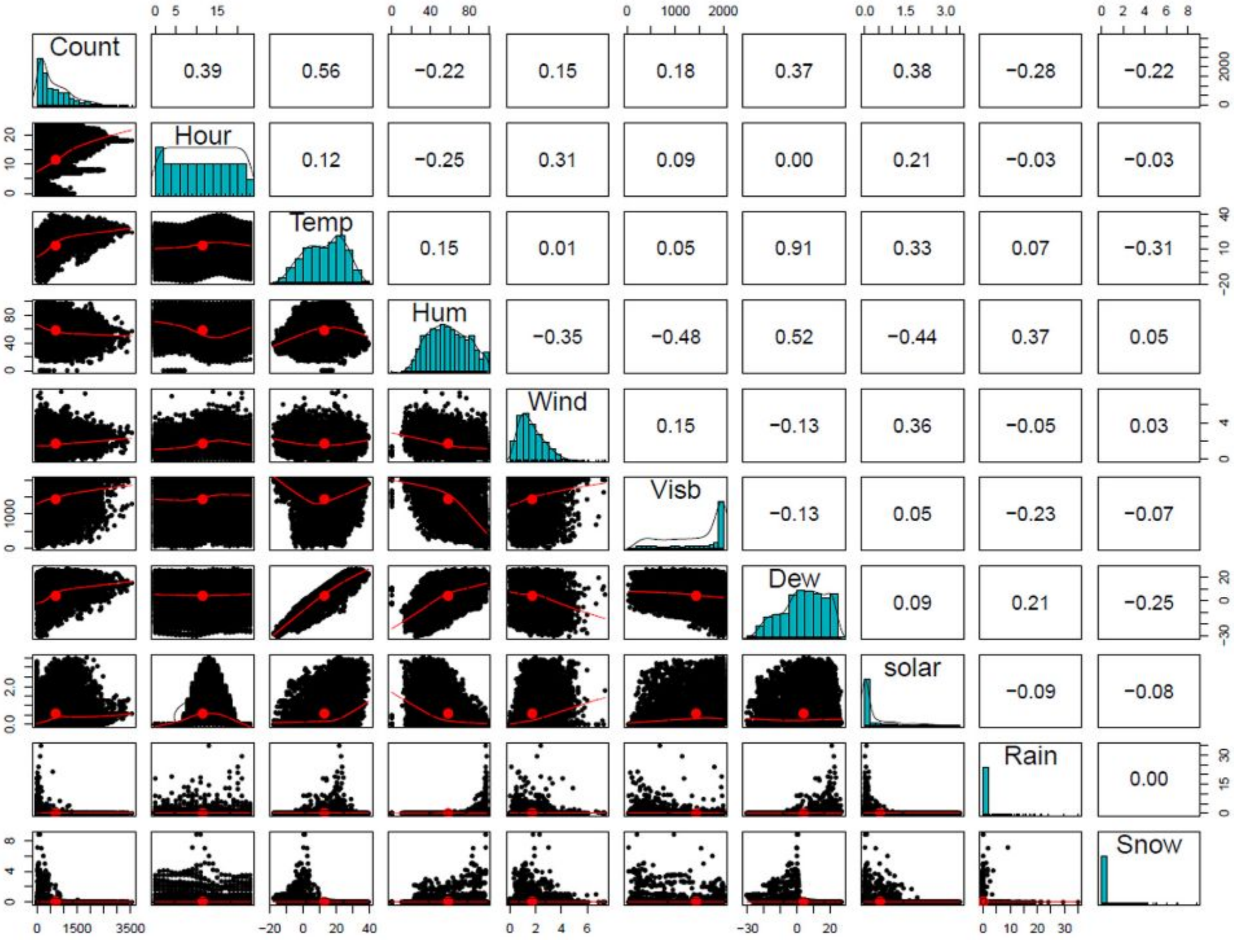
Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data

Study correlations and relationships between data.

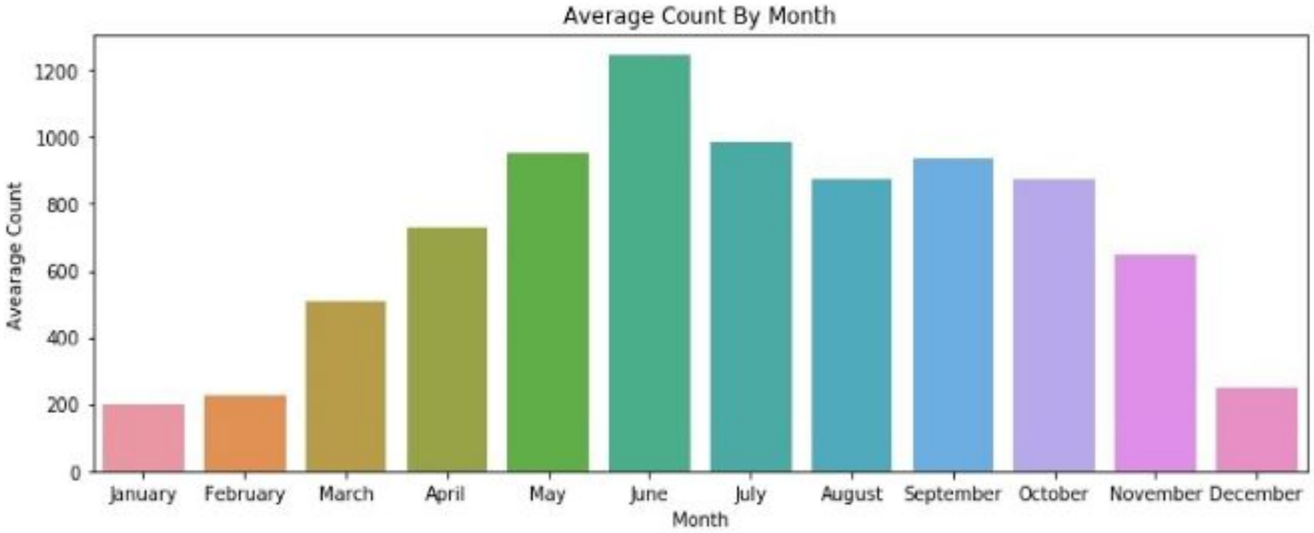
Pearson correlation coefficient:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$



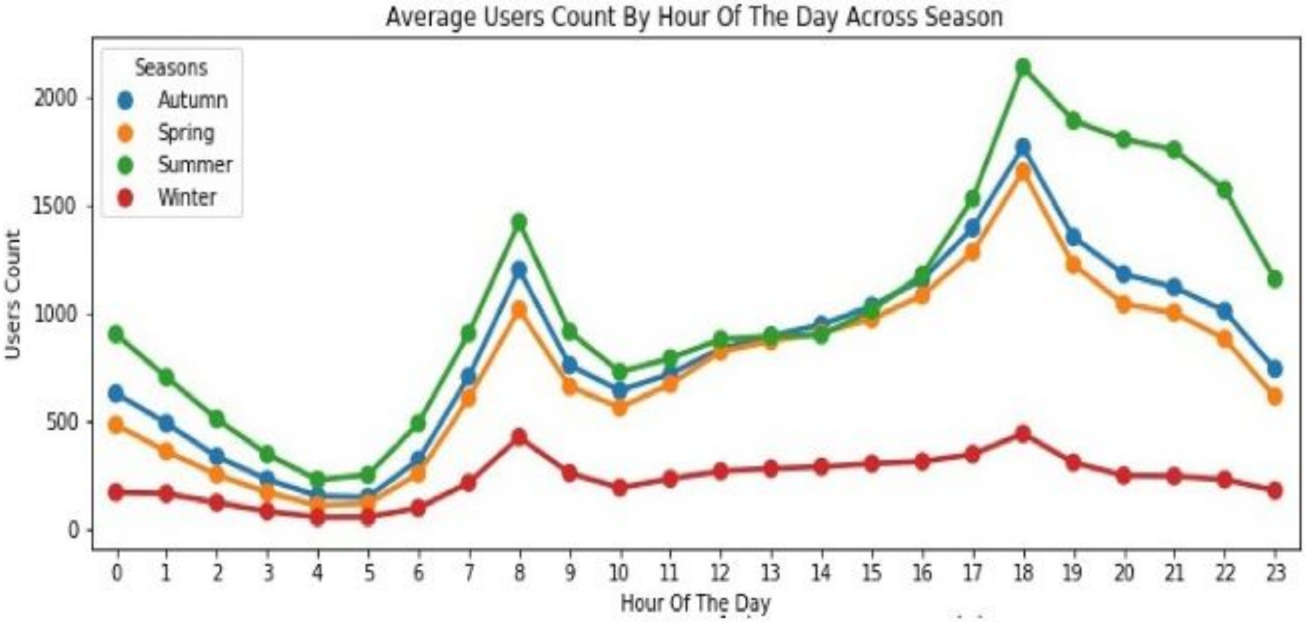
Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data



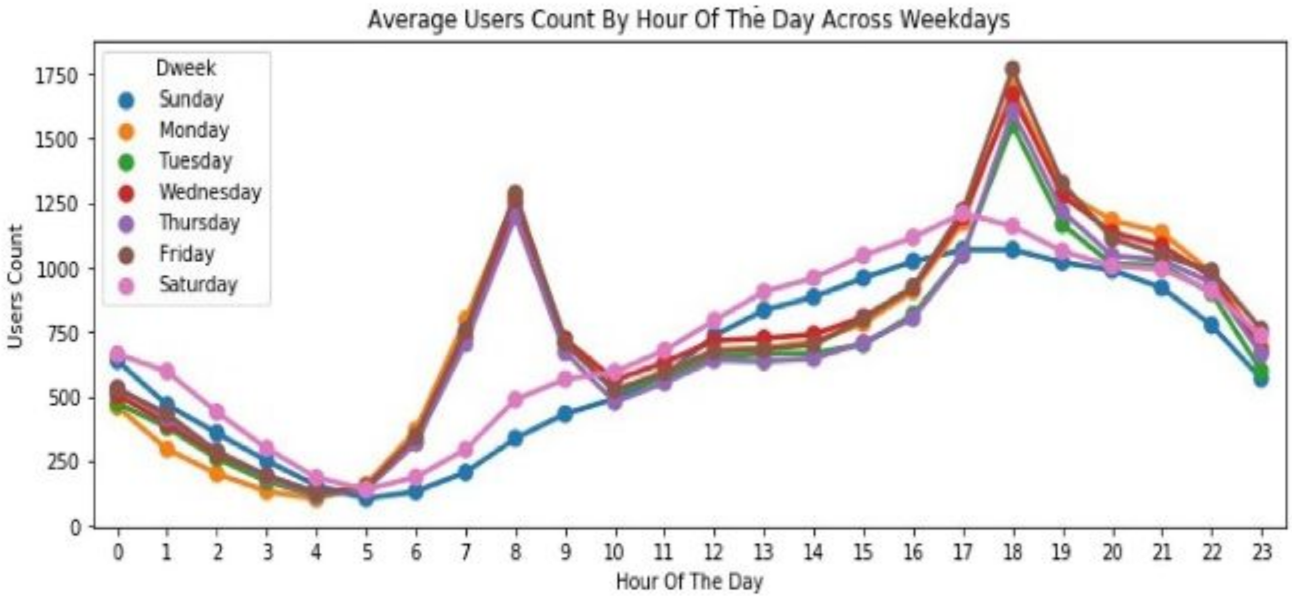
Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data



Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore the Data



Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Explore

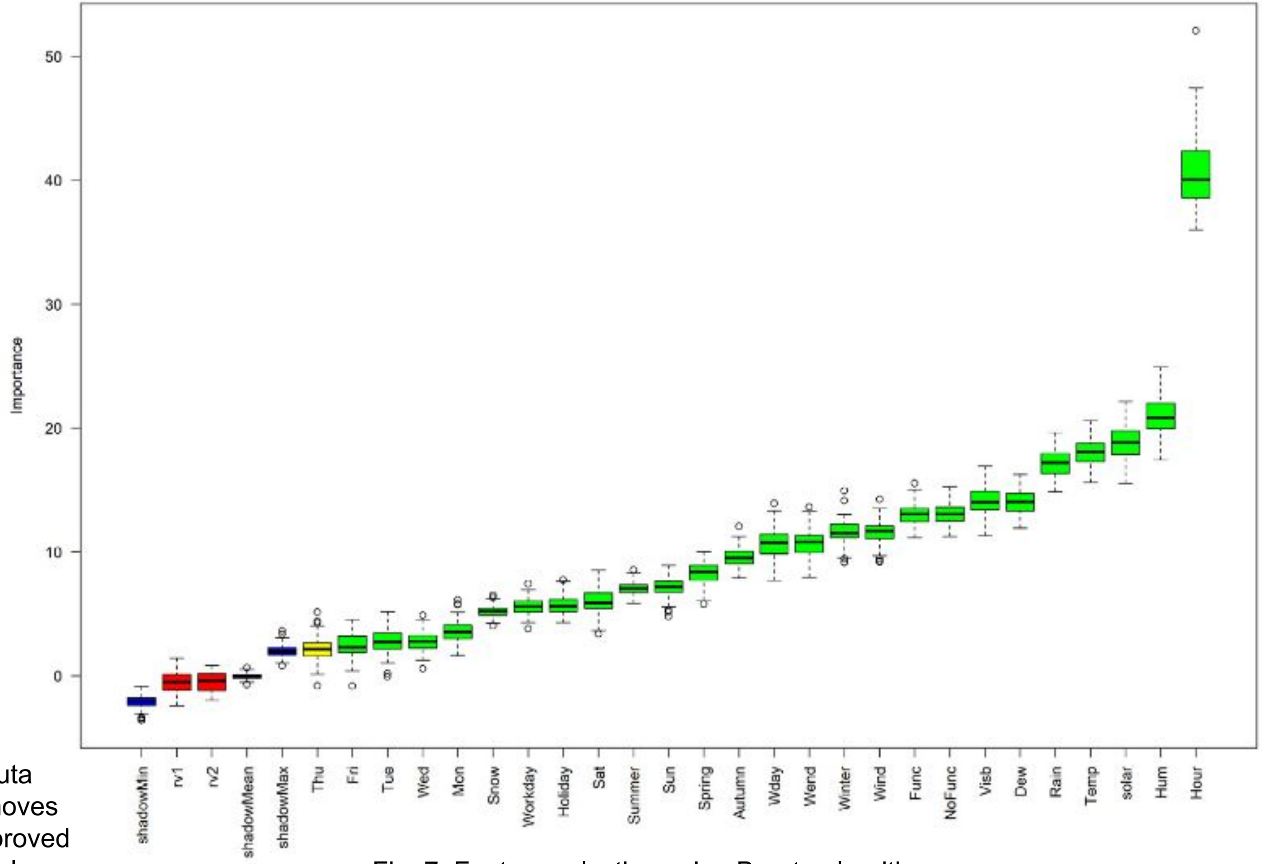


Fig. 7. Feature selection using Boruta algorithm.

[not examined] The Boruta algorithm iteratively removes the features which are proved by a statistical test to be less relevant than random probes

Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." *Computer Communications* 153 (2020): 353-366.

Kursa, M. B., & Rudnicki, W. R. (2010). Feature Selection with the Boruta Package. *Journal of Statistical Software*, *36*(11), 1–13. https://doi.org/10.18637/jss.v036.i11

Prepare the Data

- Handle outliers / wrong data
    - E.g. Lat / Long set to 999 in a dataset I work with.

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?
  - Impute missing data (average? Predict?)

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?
  - Impute missing data (average? Predict?)
- Feature engineering
  - Convert categorical data to one-hot encoding

Hire Purpose
Shopping
Shopping
Commute
See friends
See friends
Commute
Sightseeing
Commute

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?
  - Impute missing data (average? Predict?)
- Feature engineering
  - Convert categorical data to one-hot encoding

Hire Purpose			
Shopping	**Commute**	**SeeFriends**	**Sightseeing**
1	0	0	0
1	0	0	0
0	1	0	0
0	0	1	0
0	0	1	0
0	1	0	0
0	0	0	1
0	1	0	0

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?
  - Impute missing data (average? Predict?)
- Feature engineering
  - Convert categorical data to one-hot encoding
  - Convert a continuous feature to a categorical one

Prepare the Data

- Handle outliers / wrong data
    - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
    - Drop those rows? (does this cause a bias?)
    - Drop whole feature?
    - Impute missing data (average? Predict?)
- Feature engineering
    - Convert categorical data to one-hot encoding
    - Convert a continuous feature to a categorical one
    - Add new features computed from old ones

Prepare the Data

- Handle outliers / wrong data
  - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
  - Drop those rows? (does this cause a bias?)
  - Drop whole feature?
  - Impute missing data (average? Predict?)
- Feature engineering
  - Convert categorical data to one-hot encoding
  - Convert a continuous feature to a categorical one
  - Add new features computed from old ones
  - E.g. compute ‘distance to nearest big road’ from lat/long – as this will be useful for air pollution

Prepare the Data

- Handle outliers / wrong data
    - E.g. Lat / Long set to 999 in a dataset I work with.
- Handle missing data
    - Drop those rows? (does this cause a bias?)
    - Drop whole feature?
    - Impute missing data (average? Predict?)
- Feature engineering
    - Convert categorical data to one-hot encoding
    - Convert a continuous feature to a categorical one
    - Add new features computed from old ones
    - E.g. compute ‘distance to nearest big road’ from lat/long – as this will be useful for air pollution
    - Normalise…

Normalising

- Why does it matter?
    - Consider this example: We want to classify which child is malnourished using their age and height.

Weight (in kg)	Age (in years)	Malnourished?
9.1	1.1	No
9	1.1	No
5	0.5	Yes
8	1.5	No
6.1	0.9	Yes
9.2	1.5	No
18.3	1.9	No

Normalising

- Why does it matter?
    - Consider this example: We want to classify which child is malnourished using their age and height.

Weight (in kg)	Age (in years)	Malnourished?
9.1	1.1	No
9	1.1	No
5	0.5	Yes
8	1.5	No
6.1	0.9	Yes
9.2	1.5	No
18.3	1.9	No

    - The problem is the nearest neighbour is driven mostly by the weight, and age is ignored.

Normalising

- Why does it matter?
  - Consider this example: We want to classify which child is malnourished using their age and height.

Weight (in kg)	Age (in years)	Malnourished?
9.1	1.1	No
9	1.1	No
5	0.5	Yes
8	1.5	No
6.1	0.9	Yes
9.2	1.5	No
18.3	1.9	No

  - The problem is the nearest neighbour is driven mostly by the weight, and age is ignored.
- We have to normalise so that they are comparable distances.
  - Min-max scaling: rescaling the range to be either [0, 1] or [−1, 1].

Normalising

- Why does it matter?
  - Consider this example: We want to classify which child is malnourished using their age and height.

Weight (in kg)	Age (in years)	Malnourished?
9.1	1.1	No
9	1.1	No
5	0.5	Yes
8	1.5	No
6.1	0.9	Yes
9.2	1.5	No
18.3	1.9	No

  - The problem is the nearest neighbour is driven mostly by the weight, and age is ignored.
- We have to normalise so that they are comparable distances.
  - Min-max scaling: rescaling the range to be either [0, 1] or [−1, 1].
  - Standardisation (also called z-score normalisation): subtract mean and divide by standard deviation.

(min-max has particular problems if there are outliers).

Shortlist/Try Models

- There might be a lot of choices (e.g. all the preprocessing we just discussed, plus a range of models, each with a range of hyper parameters).
    - Random / grid search hyperparameters.
    - Could use Auto-ML to support this.
- Compare performance using the same training/validation data split! (otherwise it's not comparable).

- Finally, pick the model and preprocessing etc that seem best and fit on the whole training/validation set.
- Test on your held-out test data.

Do not try to change your model based on the performance on the test data!

Take Home Message

- Most of applied ML is:
  - Feature engineering
  - Data cleaning
  - Learning about your data and working with those who will use your predictions.
- Take care selecting how you will evaluate your model:
  - If data isn’t balanced, or you care more about a FN than a FP, take this into account.

True Class \ Predicted Class	Positive	Negative
Positive	6	14
Negative	10	70