

# Time Series Forecasting: Population levels in Austin Animal Shelter

**Helen Skinner**

# Contents

- Problem and solution
- Data
- Model
- Results
- Uses, shortcomings and further work
- Acknowledgements

# Problem and solution

**Problem:** For resourcing and budgeting reasons, animal shelters need to accurately forecast the animal population.

## **Solution:**

- Build a time series model to forecast the population ahead 1 year
  - Ensure it can be re-run with updated data as required
- Aim to make the model as accurate as possible

# Data – Overview

**data.austintexas.gov**



October 1, 2013 to  
present day



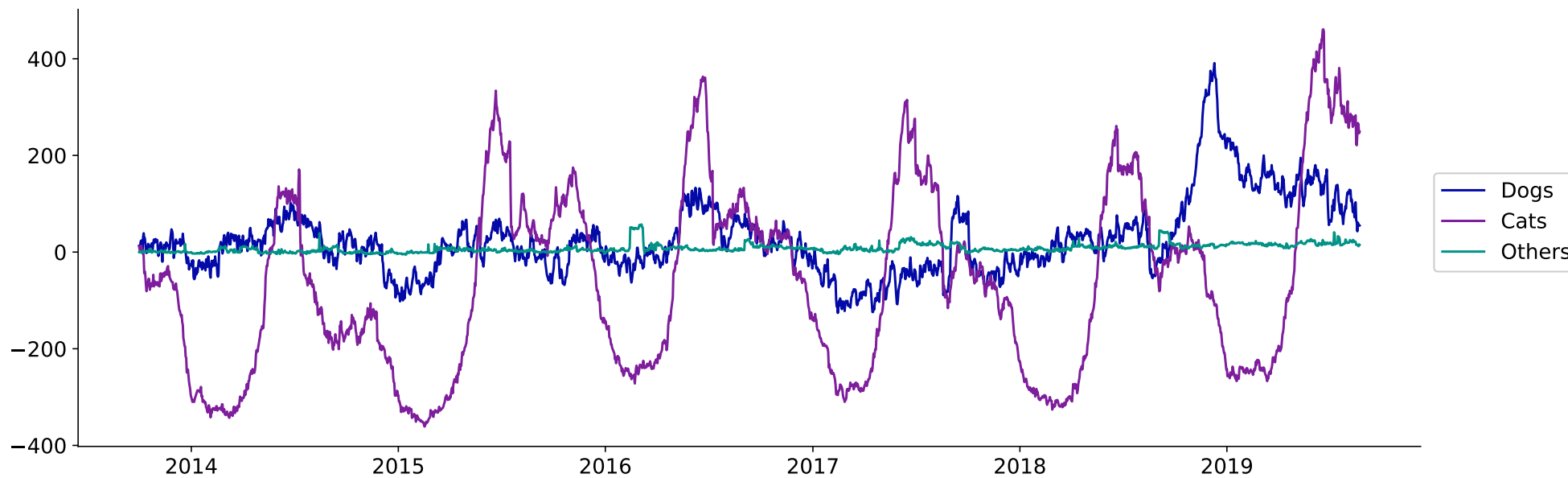
5 animal categories:  
dogs, cats, birds,  
livestock and other

## **Data preprocessing required:**

- Convert intake/outtake to population
- Group birds, livestock and other into single category
- Create time series variables where required

# Data – Animal population

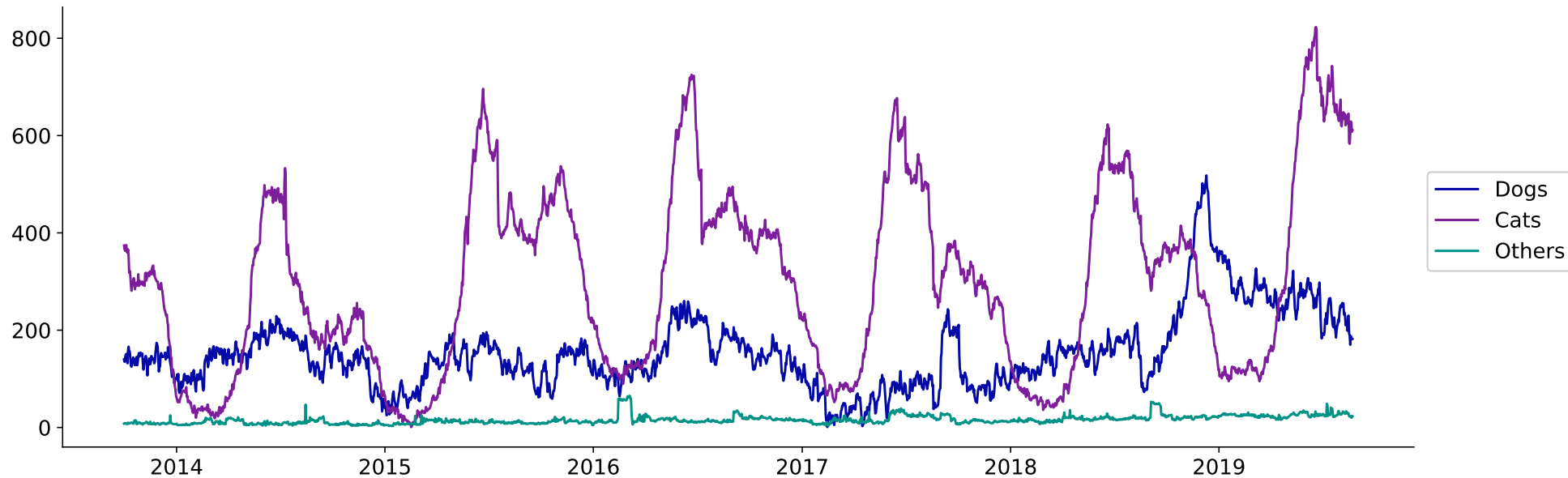
Shelter population changes from October 1, 2013



- Annual seasonality for cats
  - Maybe for dogs too
  - Potential weekly seasonality but hard to see
- Divide by zero errors could be an issue

# Data – Animal population

Shelter population from October 1, 2013 - rebased



- Rebased population data so never goes to zero
- Potential increasing trend visible?

# Unsupervised learning - Seasonality

**Aim:** Use unsupervised clustering techniques to understand what seasonality exists within the data.

## **How:**

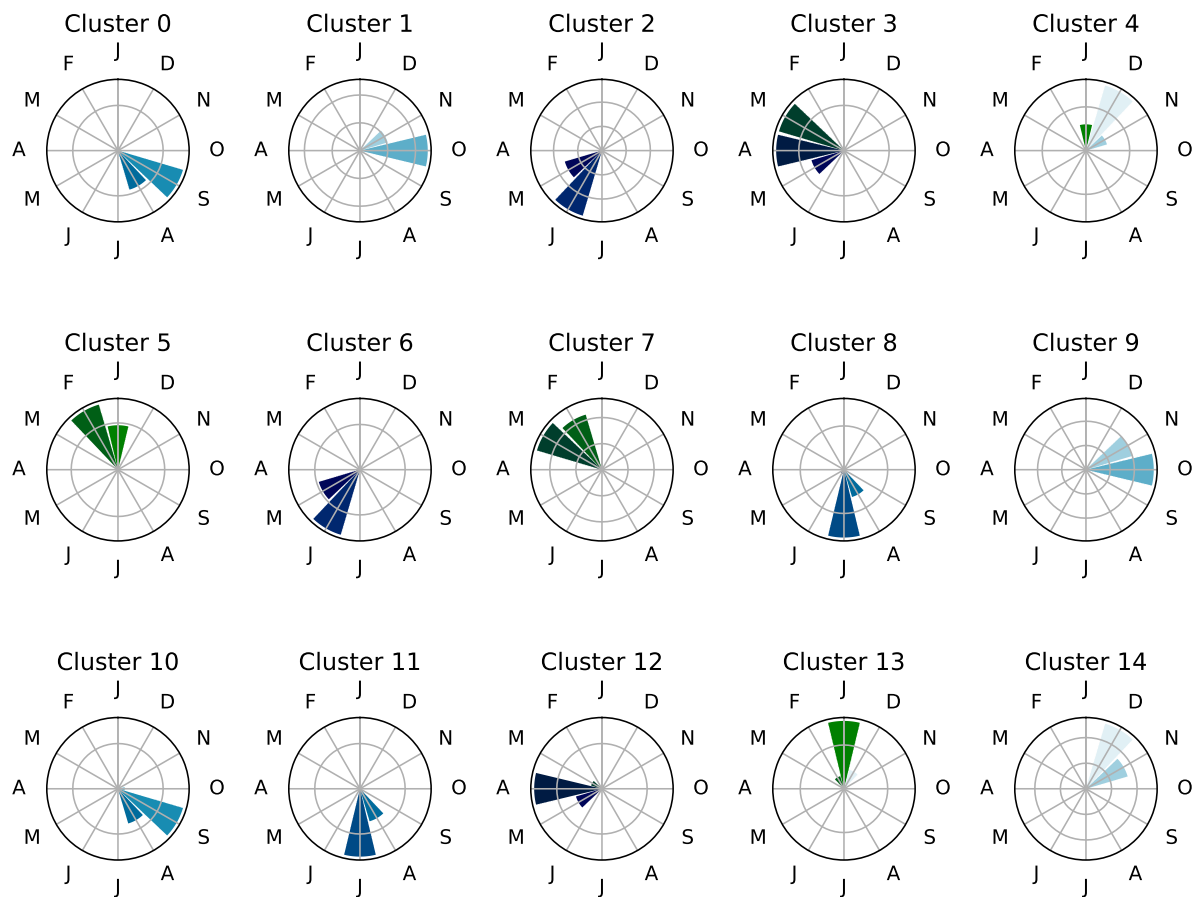
- Use a KMeans algorithm
  - Flexible (can try different numbers of clusters)
  - Fast
- Look to see if months / days group together indicating similarity



# Unsupervised learning - Dogs

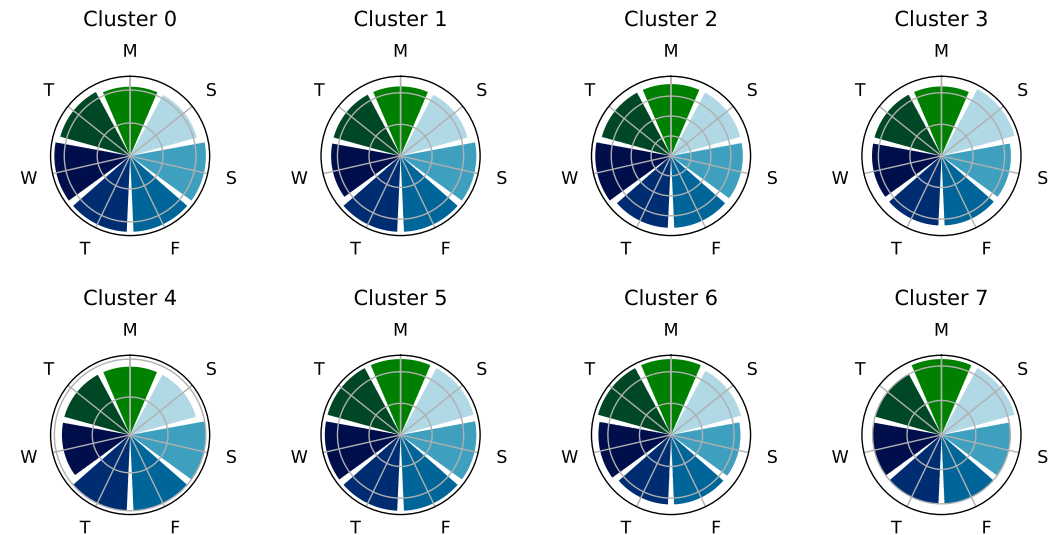
## Clusters found with KMeans - Dogs | Months

Similar months are grouped together indicating potential annual seasonality



## Clusters found with KMeans - Dogs | Days

Days are spread across all clusters indicating no relationship



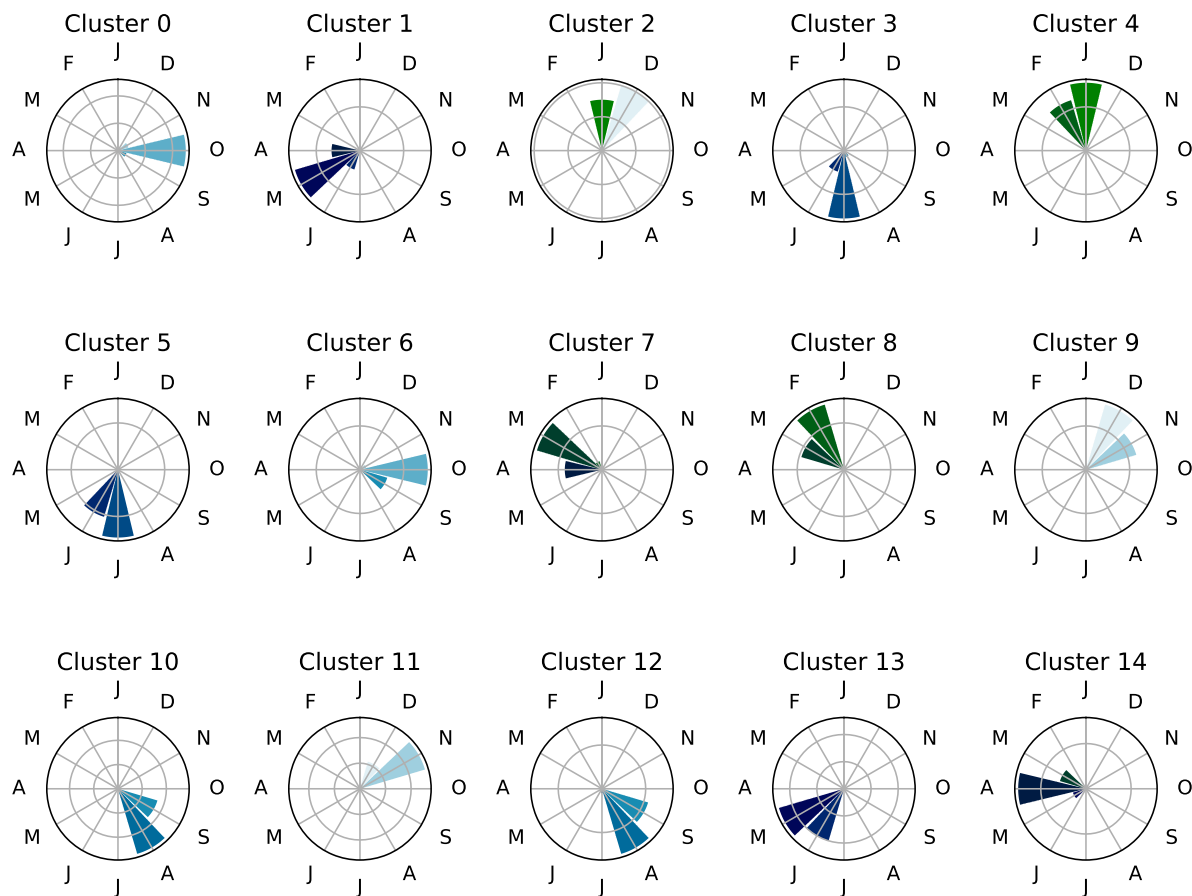
- No visible grouping by day
- Clear grouping by month
  - Not exact month but same time of year



# Unsupervised learning - Cats

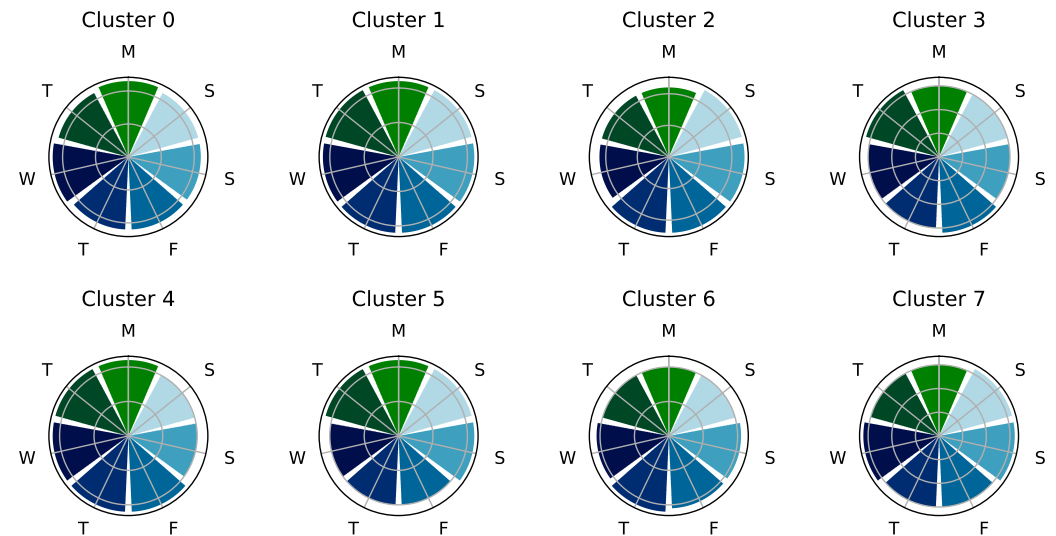
## Clusters found with KMeans - Cats | Months

Similar months are grouped together indicating potential annual seasonality



## Clusters found with KMeans - Cats | Days

Days are spread across all clusters indicating no relationship

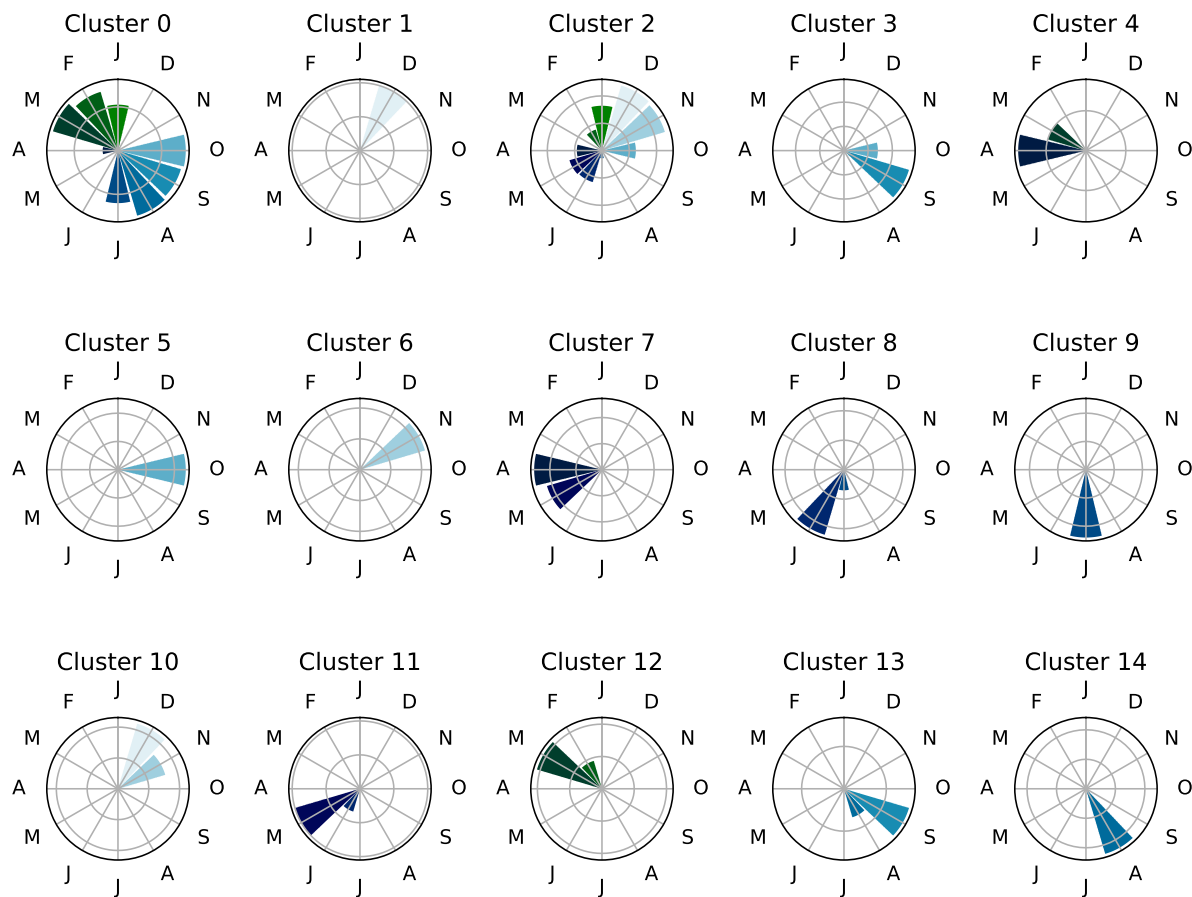


- No visible grouping by day
- Clear grouping by month
  - Not exact month but same time of year

# Unsupervised learning - Others

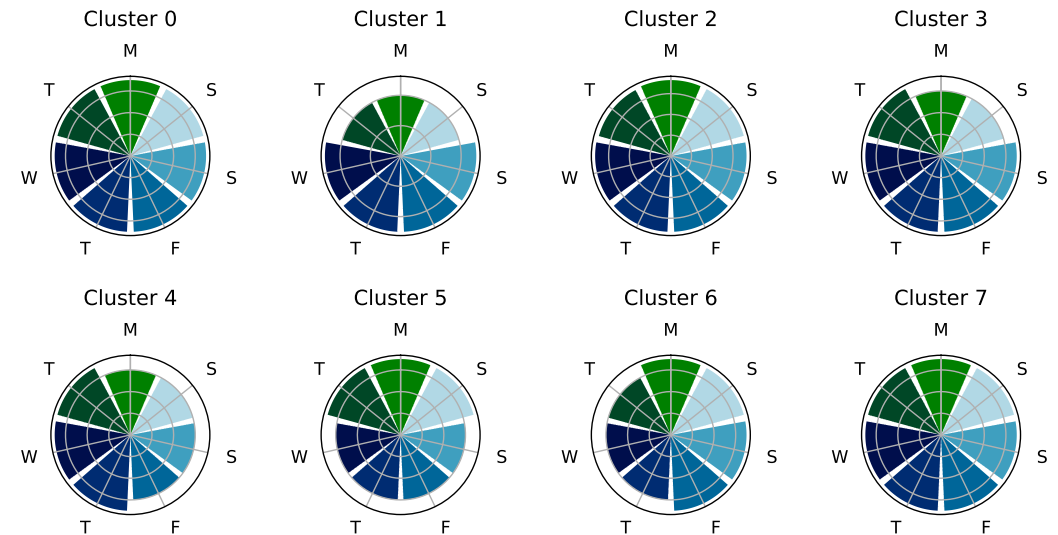
## Clusters found with KMeans - Others | Months

Similar months are grouped together indicating potential annual seasonality



## Clusters found with KMeans - Others | Days

Days are spread across all clusters indicating no relationship



- No visible grouping by day
- Limited grouping by month
  - Concentrated in cluster 0 and cluster 2

# Modeling – Algorithms tested

## Baseline algorithms

### Naive

Assumes future values will be equal to the most recent known value.

### Seasonal Naive

Assumes future values will be equal to the average of the known values at the same date.

## Time series algorithms

### ARIMA

Assumes future values are based on previous terms and previous errors.

### Prophet

Algorithm developed by Facebook which separately models trend and seasonality.

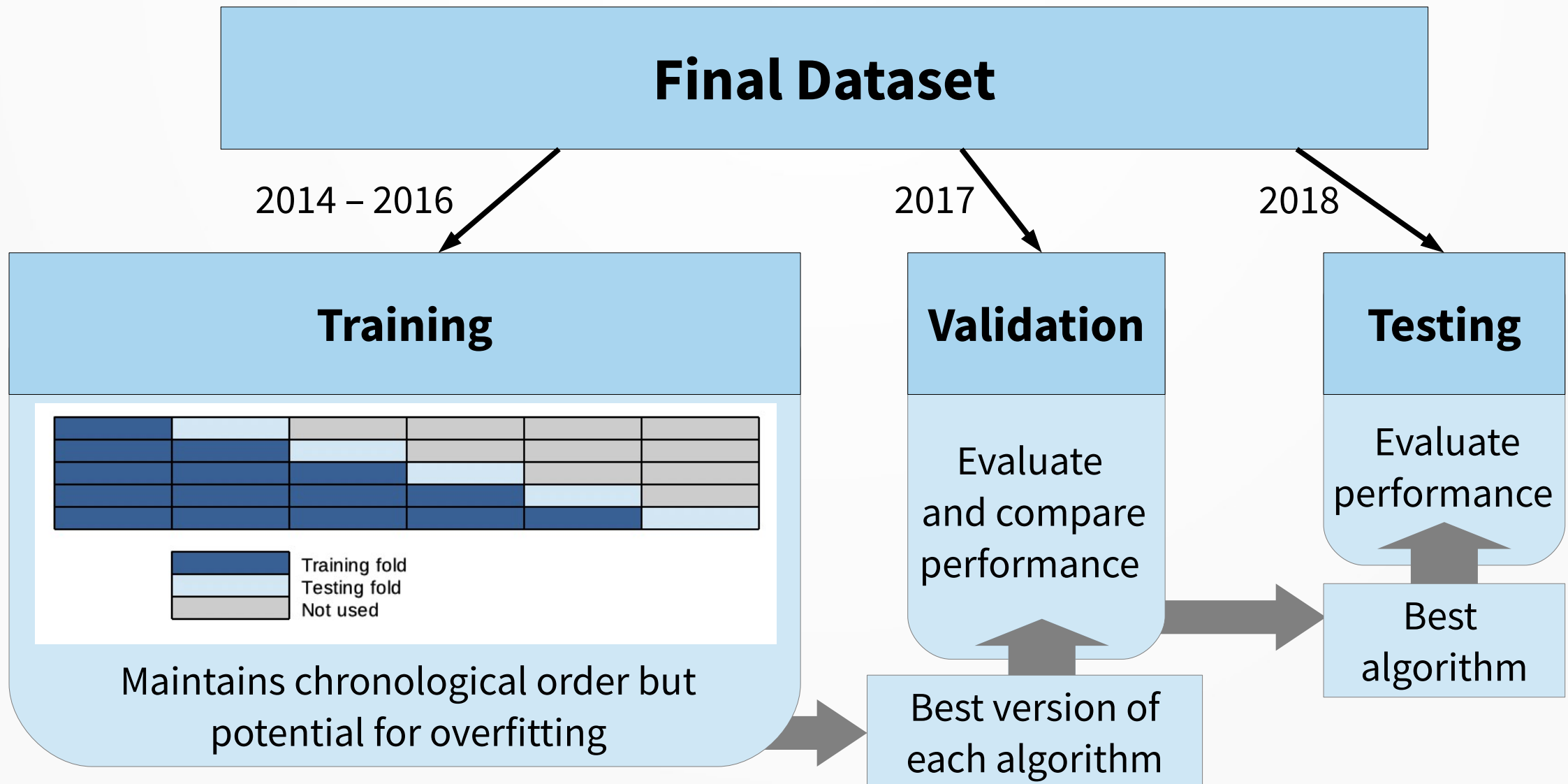
### Neural Network

Supervised learning technique which uses previous terms as features to learn from.

### Random Forest

Supervised learning technique which uses previous terms as features to learn from.

# Modeling – Selection pathway



# Modeling – Error evaluation

## MAE

$$\text{mean}(|e_t|)$$

- Mean Absolute Error
- Same scale as data
  - Cannot use to make comparisons between different time series
- Minimizing this leads to forecasts of the median

## MAPE

$$\text{mean}(|p_t|)$$

- Mean Absolute Percentage Error
- Unit free so can compare between different time series
- Gives extreme values when  $y_t$  is close to zero and is undefined if  $y_t$  is equal to zero

$$p_t = \frac{100 e_t}{y_t}$$

## RMSE

$$\sqrt{\text{mean}(e_t^2)}$$

- Root Mean Squared Error
- Same scale as data
  - Cannot use to make comparisons between different time series
- Minimizing this leads to forecasts of the mean

## MASE

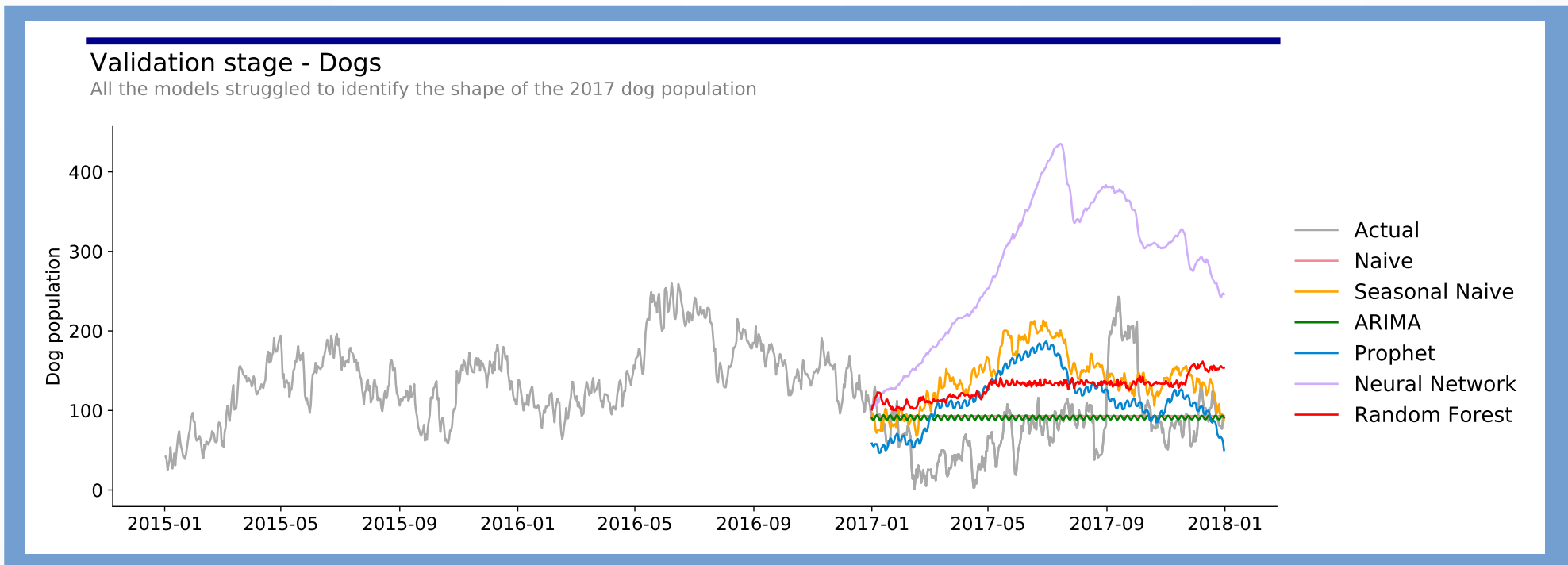
$$\text{mean}(|q_j|)$$

- Mean Absolute Scaled Error
- Can compare between different time series
- Uses a scaled error based on a naive forecast of the training data

$$q_j = \frac{e_j}{\frac{1}{T-1} \sum_{t=2}^T |y_t - y_{t-1}|}$$



# Modeling – Dog validation results

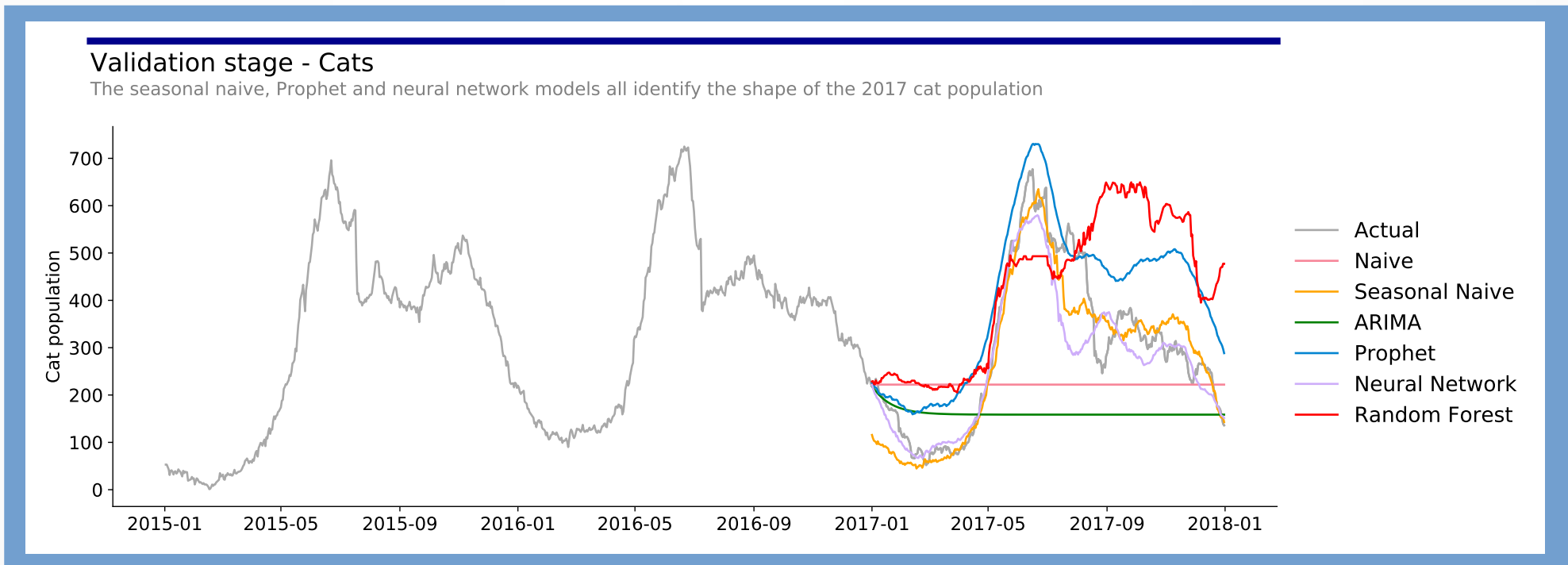


Baselines	MAE	RMSE	MAPE	MASE
Naive	35.5	48.2	125.9	4.2
Seasonal Naive	72.8	82.1	205.1	8.6

TS Models	MAE	RMSE	MAPE	MASE
ARIMA	34.7	47.8	121.2	4.1
Prophet	56.8	66.1	149.4	6.7
Neural Network	201.0	215.9	439.9	23.8
Random Forest	59.6	64.6	181.9	7.1



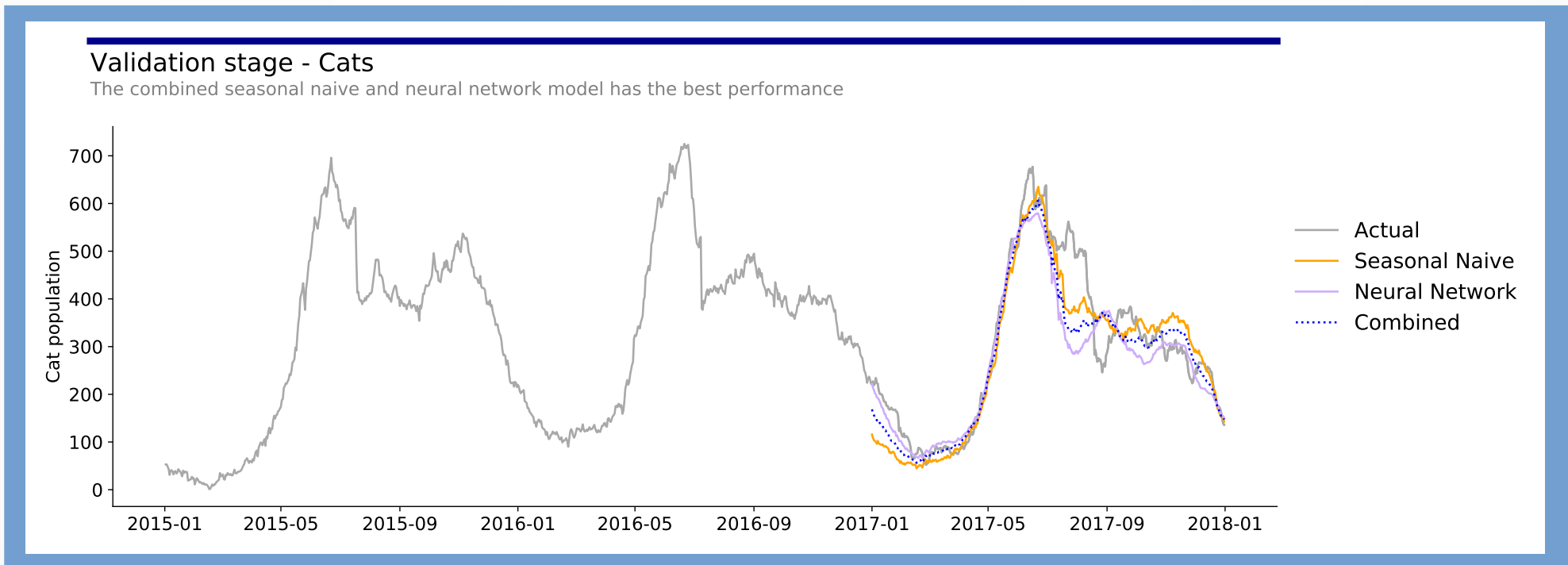
# Modeling – Cat validation results



Baselines	MAE	RMSE	MAPE	MASE
Naive	145.7	183.6	63.2	16.9
Seasonal Naive	44.3	58.9	17.9	5.1

TS Models	MAE	RMSE	MAPE	MASE
ARIMA	170.2	217.8	57.0	19.7
Prophet	108.2	123.0	55.3	12.5
Neural Network	47.5	71.8	16.0	5.5
Random Forest	158.4	188.8	79.6	18.3

# Modeling – Cat validation results 2



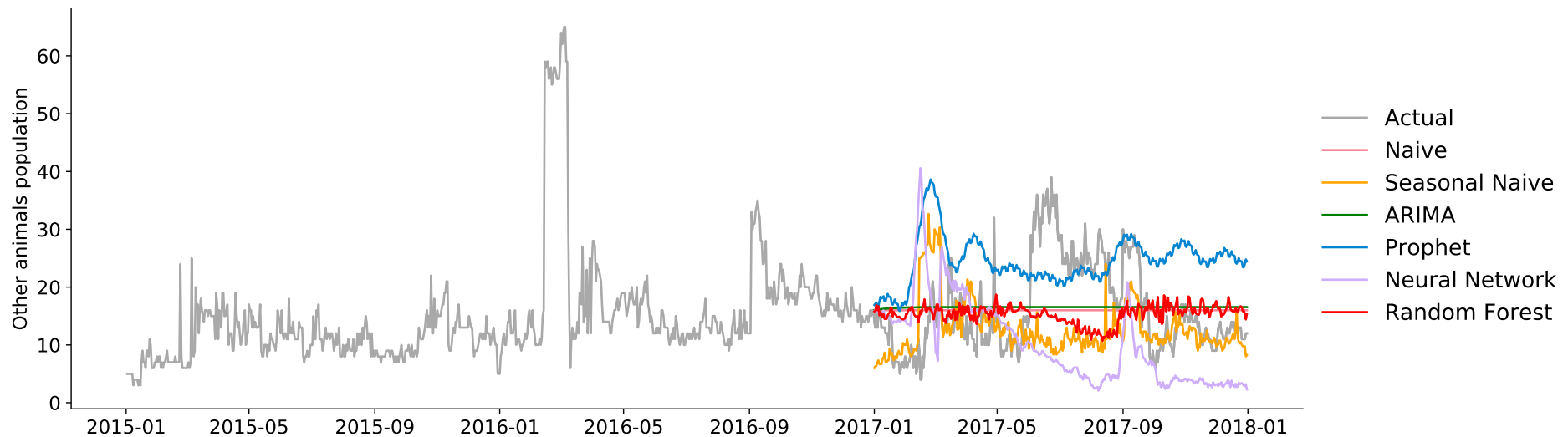
Baselines	MAE	RMSE	MAPE	MASE
Naive	145.7	183.6	63.2	16.9
Seasonal Naive	44.3	58.9	17.9	5.1

TS Models	MAE	RMSE	MAPE	MASE
ARIMA	170.2	217.8	57.0	19.7
Prophet	108.2	123.0	55.3	12.5
Neural Network	47.5	71.8	16.0	5.5
Random Forest	158.4	188.8	79.6	18.3
Combined	42.5	61.2	15.2	4.9

# Modeling – Other validation results

## Validation stage - Others

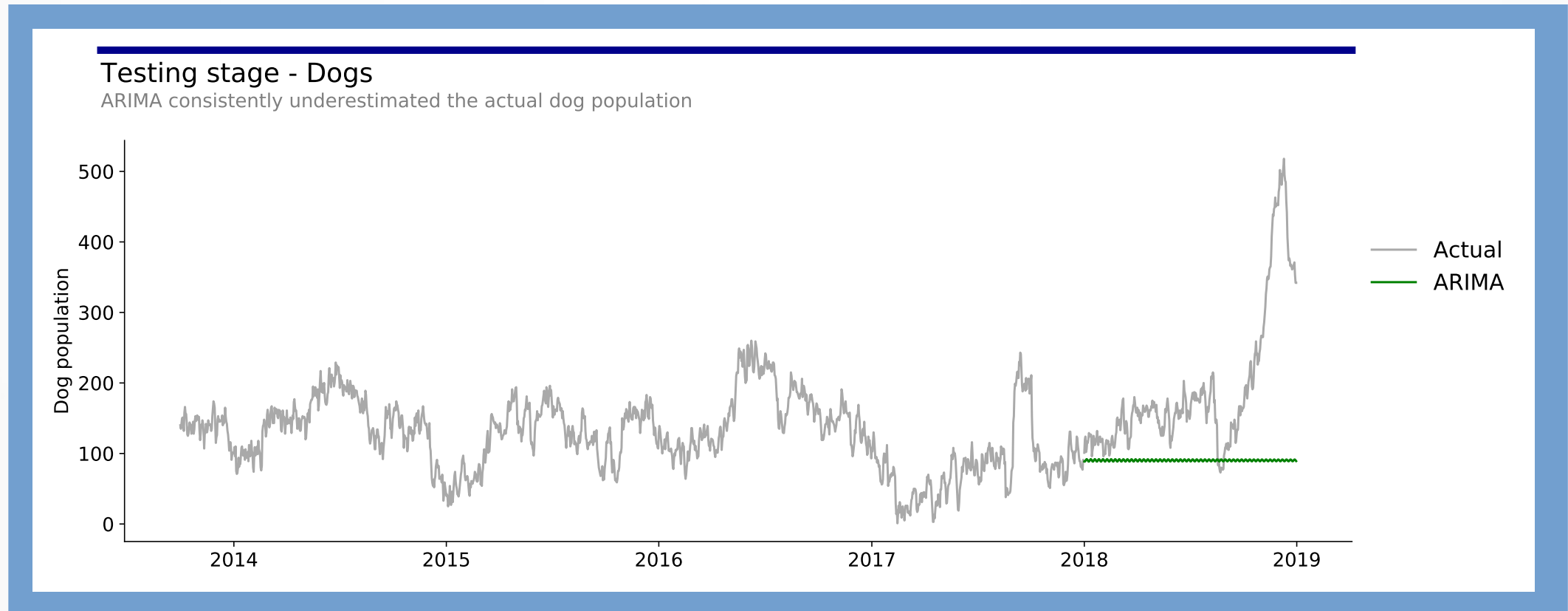
All the models struggled to identify the shape of the 2017 'other animals' population



Baselines	MAE	RMSE	MAPE	MASE
Naive	6.3	7.7	45.2	3.3
Seasonal Naive	7.1	9.9	42.0	3.8

TS Models	MAE	RMSE	MAPE	MASE
ARIMA	6.4	7.7	47.5	3.4
Prophet	10.7	12.2	93.3	5.7
Neural Network	10.7	13.2	70.6	5.7
Random Forest	6.9	8.5	47.2	3.6

# Modeling – Dog test results



Test  
*Validation*

**MAE**  
131.1  
*34.7*

**RMSE**  
159.1  
*47.8*

**MAPE**  
52.3  
*121.2*

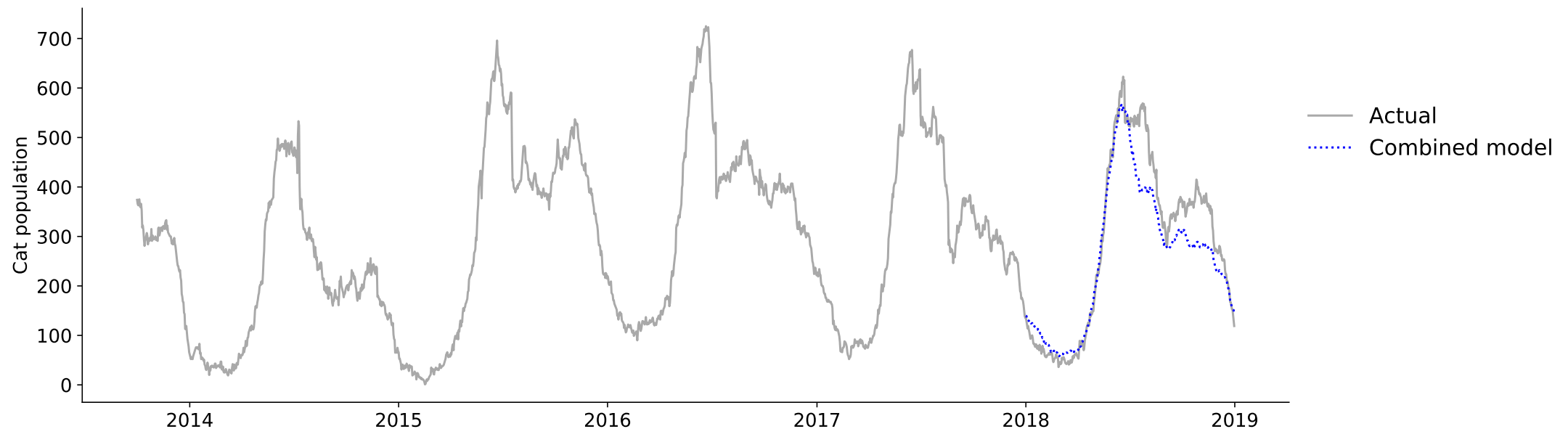
**MASE**  
14.9  
*4.1*

- Forecasts almost constant value
  - Misses upward trend
- Spike is unusual – perhaps unpredictable

# Modeling – Cat test results

## Testing stage - Cats

The combined model has accurately modeled the behavior of the cat population



Test  
*Validation*

**MAE**

88.0

42.5

**RMSE**

125.1

61.2

**MAPE**

27.6

15.2

**MASE**

9.8

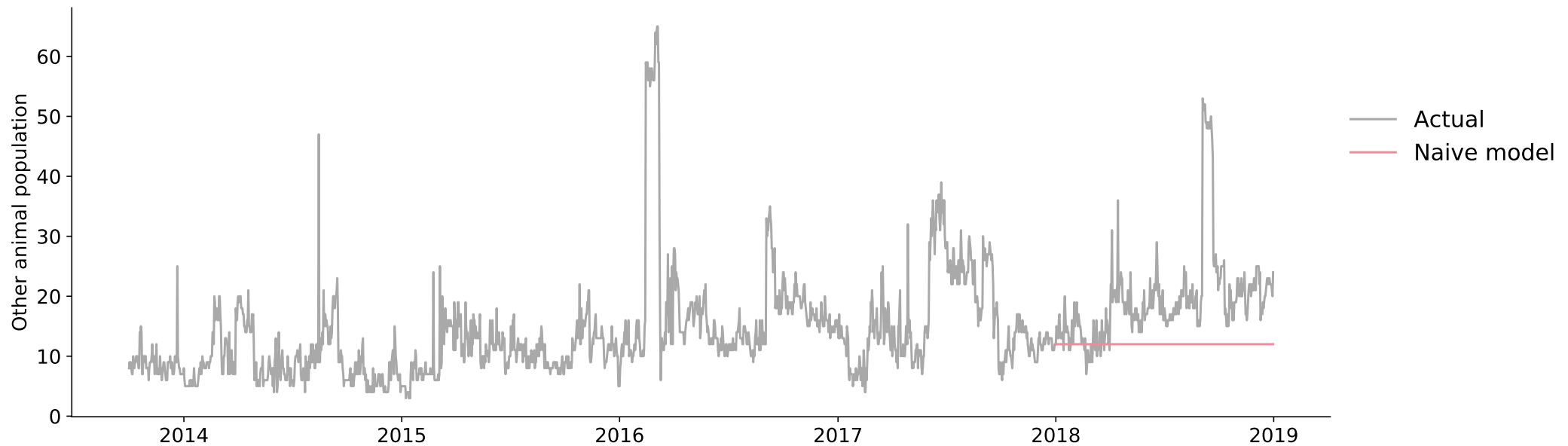
4.9

- Tracks seasonal pattern very well
  - Misses secondary peak
- Overall slightly underestimates

# Modeling – Other test results

## Testing stage - Other

The naive model has underestimated the other animal population



Test  
*Validation*

**MAE**

10.1

6.3

**RMSE**

12.4

7.7

**MAPE**

41.0

121.2

**MASE**

5.4

4.1

- Forecasts constant value
  - Misses upward trend
- Spikes appear random



# Strengths of the models

- Cat model
  - Accurately identifies the general shape of the population
  - Animal shelter is able to estimate when peaks are likely to occur
- Dog and other models
  - Models are more ‘boring’ but have lower errors, so less likely to be as significantly wrong as more complex models
  - Reasonable short term estimates but quickly outdated
    - Potentially could improve by re-running models more often

# Weak points of the models

- Evaluation metrics
  - ‘Averaging’ models have lowest errors but forecast straight lines
  - Dog and other models not useful for capturing expected up and down periods
  - Potentially alternative evaluation metrics are required
- Random spikes
  - Almost impossible to predict
- Overfitting
  - Very likely with sliding windows cross validation
  - Difficult to combine the need for long training data periods and the decreasing usefulness of older data

# Further work

## **Additional data**

- Employment / economic data
- Animal health – perhaps using NLP techniques
- Intervention campaigns e.g. discounted adoption fees

## **Evaluation metrics**

- Consider what desired outcome is
- Is underestimating worse than overestimating?

## **Different data**

- Intake / Outtake
- The daily change (rather than absolute daily number)

## **Further hyperparameter tuning**

- Prophet – holidays, specific changepoints
- NN / RF – several more options to consider

## **Further feature engineering**

- Understand feature importances (look back periods, months / days)
- Use PCA to remove collinearity

## **Different models**

- RNNs / LSTMs

# Acknowledgements

- Technical advice and support gratefully received from
  - Jenny Yu
  - Tom Nickson
  - Technical Coaching and peer group via Slack
- References
  - [Forecasting: Principles and Practice](#) by Rob J Hyndman and George Athanasopoulos
- Icons
  - Icons made by [OCHA](#) and [Freepik](#) from [www.flaticon.com](http://www.flaticon.com)