REGRESSION OF ANNUAL BUSHFIRE SEVERITY ON AGGREGATED RAINFALL MEASURES

Charles O'Neill
College of Science
Australian National University
Canberra, ACT 2601
charlie@macuject.com

December 1, 2021

ABSTRACT

The given dataset consists of monthly rainfall in millimetres in the Sydney catchment area from 1960 to 2020, as well as the total hectares burnt in bushfires in the same period. Whilst we initially (naïvely) treat rainfall in the relevant year as the predictor variable and the hectares burnt as as the variable to be forecast, we will proceed to more complicated time-series methods. In particular, we will utilise models that rely on data not just from the current period, but previous time-steps, and place appropriate weightings on the remoteness of such time-steps. The final model structure will involve feeding in several time-steps, each with monthly rainfall in a twelve-dimensional vector, and previous year bushfire data as a scalar in each step. Whilst the cessation of aggregation reduces the explainability of the model, doing so gives the recurrent neural network as much flexibility as possible to find the most complex relationships between monthly rainfall and bushfire severity.

1 Introduction

The goal is to predict total hectares burnt in bushfires in the coming year, given rainfall information from current and previous years, and bushfire information from previous years. We divide the data into 75% training data, 25% testing data. This train-test split is not random, but rather segmented at a specific year in the time-series, to ensure the model's performance does not appear improperly successful due to the time-series interpolation problem.[1] We will use two metrics to quantify the strength of our models: root mean squared error on the test-set predictions, and the coefficient of determination R^2 on the model fit on the whole dataset. These aim to capture the accuracy and explainability of the model, respectively.

2 Initial data analysis

2.1 Descriptive statistics

A summary of data statistics is shown in Table 1.

2.2 Correlations

Clearly, defining the Dennison index as aggregated rainfall from October to January gives a stronger linear relationship (as measured by correlation coefficient), than other aggregate rainfall measures, with hectares burnt in bushfires. However, there are several caveats to keep in mind with this definition:

[1] In time-series data, if the testing-data is taken as random samples from throughout the series, then the model can learn to interpolate the required value by looking at the previous and next value. This 'cheating' makes the model appear better than it actually is.

Table 1: Descriptive statistics for bushfire data

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
Aug	60	86.7	100.2	0.0	22.2	137.7	482.6
Sep	60	64.1	54.0	0.2	24.7	89.4	226.2
Oct	60	81.3	72.9	0.6	32.1	93.0	285.0
Nov	60	103.1	95.2	10.4	41.3	127.5	517.2
Dec	60	81.8	61.0	1.6	39.5	99.5	279.1
Jan	60	110.9	79.8	5.6	50.3	153.3	385.0
Feb	60	128.6	116.0	10	53.9	154.8	631
Mar	60	140.9	95.0	8	65.4	191.6	388
Apr	60	123.5	106.3	5.2	43.0	166.5	506.6
May	60	101.0	83.7	3.0	33.5	136.2	371.4
Jun	60	141.8	106.3	4	73.1	175.9	511
Jul	60	68.7	57.7	2	25.1	104.5	282
Aug-Jan	60	527.9	204.0	167.8	360.8	659.5	1,189.6
Sep-Jan	60	441.2	166.0	148.0	313.8	539.1	949.7
Oct-Jan	60	377.1	160.9	126.2	266.8	477.8	897.2
Hectares Burnt	60	283,600.0	826,836.6	0	0	219,500	5,780,000

Table 2: Correlation coefficients for varying aggregated rainfall measures.

Aggregated rainfall measure	Correlation with annual hectares burnt
October-January	-0.28
September-January	-0.25
August-January	-0.25

- Correlation coefficient only measures the strength of a linear relationship. If the relationship between rainfall and bushfire severity is non-monotonic or strongly non-linear, the correlation coefficient is not a useful metric for the strength of the relationship.
- Here we only consider the correlation between the rainfall in the same year as the bushfire data. Obviously, we later wish to feed the model previous year data as well, and the strongest measure may not be October-January rainfall in this case. We deal with this scenario below.
- In our final modelling efforts, we dispense with aggregating rainfall and instead provide the model with several years' worth of granular rainfall data in a raw format. The Dennison index is rendered superfluous in this case.

3 Naïve least-squares linear regression

Fitting a linear regression model on rainfall only from the current annual period being examined, we generate a model with an R-squared value of 0.07.

3.1 Outliers

There were significant outliers in the number of hectares burnt. Since the purpose of a baseline regression model is to get a general understanding of the relationship between rainfall and bushfire severity, we examine the effect of removing these outliers. For values that lie outside the $1.5 \times IQR$ limits, we could cap them by replacing those observations outside the lower limit with the value of 5th percentile and those that lie above the upper limit with the value of 95th percentile. Such outliers are shown in a boxplot in Figure 1.

The adjusted R^2 improves to 0.13, with the slope coefficient for the Dennison index being -594.9. When comparing years with the same aggregated rainfall from October through January, the average predicted hectares burnt in bushfires decreases by almost 600 for every additional millimetre of rain.

The results for this model, along with the models below, are shown in Table 3. A plot of model fit is shown in Figure 2.

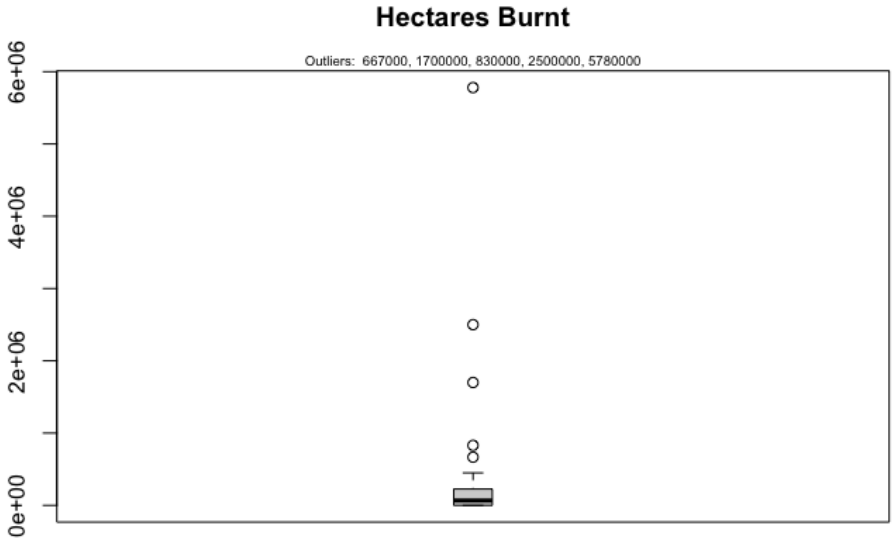

Figure 1: Outliers in the number of hectares burnt in bushfires annually.

Table 3: Regression Results

	Dependent variable:		
	Hectares Burnt in Bushfires		
	(1)	(2)	(3)
Dennison index	-594.897^{***}	$-2,806.869^{***}$	-592.077^{***}
	(187.611)	(665.450)	(194.546)
Quadratic Dennison index		2.468^{***}	
		(0.717)	
Fire last year			-0.003
			(0.095)
Fire last three years			-0.009
			(0.074)
Fire last five years			-0.017
			(0.052)
Constant	$389,461.800^{***}$	$809,756.100^{***}$	$409,590.300^{***}$
	(76,825.170)	(141,029.700)	(85,442.680)
Observations	60	60	60
R^2	0.148	0.294	0.156
Adjusted R^2	0.133	0.270	0.094

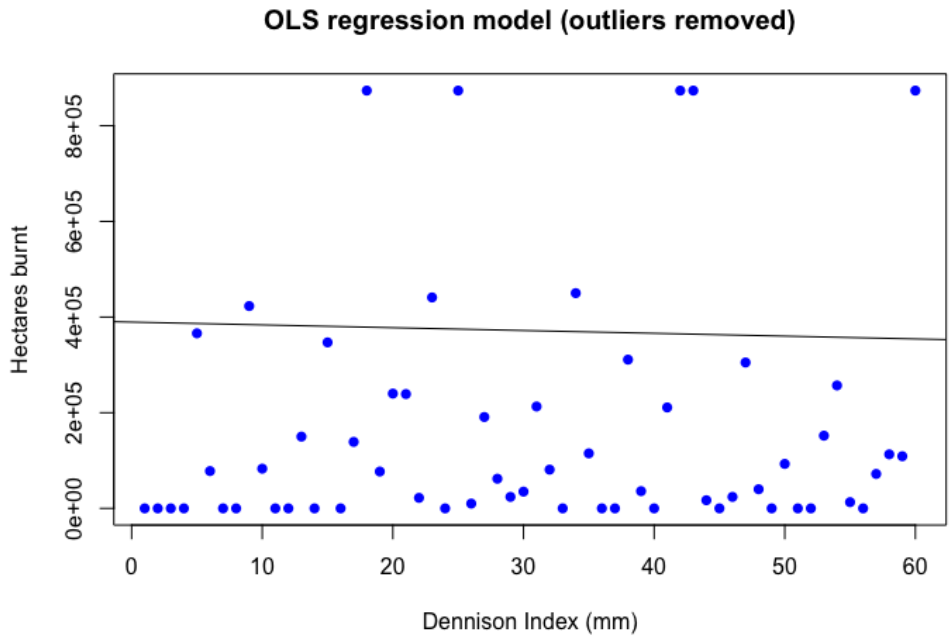

Figure 2: Simple regression model with outliers removed.

3.2 Model fit

Clearly, our low R^2 value suggests that rainfall and bushfire have a non-linear relationship, or at the least a weak linear relationship. Thus, linear regression is going to be a poor model for capturing such association.

3.3 Adding a predictor variable: previous bushfire data

Our hypothesis is that the extent of recent bushfires will have an impact on the bushfires in the current period. This is likely due to the amount of fuel available for the bushfire to consume. To test this hypothesis, we aggregate hectares burnt in the last one, three and five years, and append this as a column to our dataset. Fitting a new linear regression model, we find that not one of these predictors is significant, with other predictors present in the model. However, the Dennison index remains significant to the $\alpha = 0.01$ level.

This tells us that if previous bushfire extent does have a relationship with bushfire severity in the current period, then it most likely isn't a linear one. We require a more complicated model to uncover such a relationship.

3.4 Granularity: importance ranking of months

Of course, the Dennison index is useful as an aggregation of patterns in rainfall. However, it is also of use to see which months are significant in predicting bushfire severity. From Table 4, we see that both December and January are significant, with the model's adjusted R^2 being higher than any models with only a linear Dennison index term. If we simply fit the model only on December and January rainfall, our adjusted R^2 remains high at 0.18. This suggests to us that the Dennison index may be refined further by narrowing the months it relies on.

4 Naïve and feature-engineered random forest

4.1 Decision tree

A decision tree is a good baseline model to begin with, particularly if we are to use ensembles of trees (such as random forest models) later on. The initial decision tree, which works by choosing the split at each node which improves the r-MSE of the model the most [1], is shown in Figure 3.

Table 4: Granular Monthly Regression

	Dependent variable:
	Hectares burnt (outliers removed)
Aug	-339.652
	(302.355)
Sep	-289.551
	(574.888)
Oct	-810.366^{*}
	(432.699)
Nov	247.751
	(323.448)
Dec	$-1,687.817^{***}$
	(504.669)
Jan	-826.565^{**}
	(370.682)
Feb	283.021
	(274.597)
Mar	-32.938
	(319.594)
Apr	238.957
	(299.120)
May	536.139
	(392.686)
Jun	122.168
	(332.357)
Jul	-213.230
	(531.137)
Constant	$365,060.700^{**}$
	$(136,694.700)$
Observations	60
R^2	0.377
Adjusted R^2	0.218
Residual Std. Error	220,257.000 (df = 47)
F Statistic	2.368^{**} (df = 12; 47)
Note:	*p<0.1; **p<0.05; ***p<0.01

The top node represents the initial model before any splits have been done, when all the data is in one group. This is the simplest possible model. It is the result of asking zero questions and will always predict the value to be the average value of the whole dataset. In this case, we can see it predicts a value of 223511 for the HA burnt. Moving down and to the right, this node shows us that there were 43 years where the rainfall in January was greater than 15.9mm. The average value of our dependent variable in this group is 136232. Moving down and to the left from the initial model takes us to the records where rainfall in January was less than 15.9mm.

The bottom row contains leaf nodes, at which no further splits are made. Obviously, these leaf nodes must contain at least one sample from the dataset. The mean squared error is also shown at these leaf nodes. Whilst the nodes on the left have an MSE of 0, this is because they only contain one sample. The nodes on the right have far higher MSE due to the number of samples. Our decision tree is unbalanced.

4.2 Random forest

In order to better interpret our models, we next construct a random forest. Random forests are based on the notion that, whilst training different models on subsets of data will lead to more inaccurate models, taking the average of all these different model predictions will cause the uncorrelated errors from different models to cancel out. This procedure is known as bagging. This means that we can improve the accuracy of nearly any kind of machine learning algorithm by training it multiple times, each time on a different random subset of the data, and averaging its predictions [2].

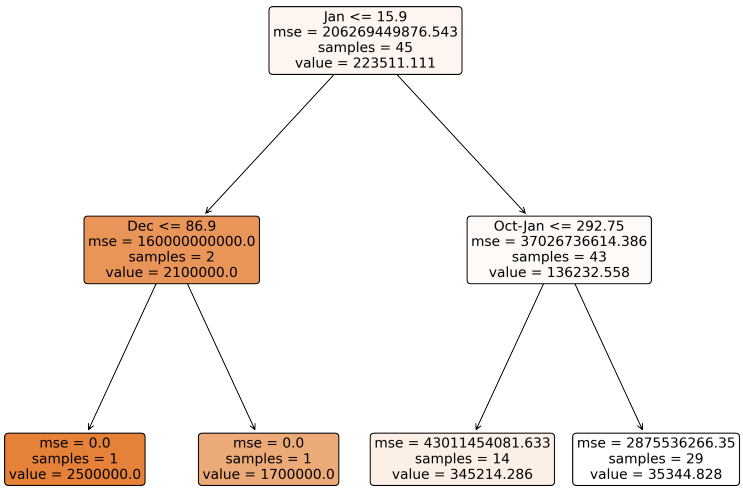

Figure 3: Initial decision tree regression model, showing that the most important factor/split is the Dennison index in August to January.

Our initial random forest with no hyperparameter tuning achieved an r-MSE of 1401525 HA on the validation set, already a marked improvement over the decision tree. However, this is still quite poor, largely due to the huge outlier of 2019-20. If we drop this year from the validation set, r-MSE drops to 285326 HA, which is certainly respectable.

4.3 Model interpretation

An advantage of random forest is that they allow for easy interpretation, and provide answers to questions such as:

- How confident are we in our predictions using a particular row of data?
- For predicting with a particular row of data, what were the most important factors, and how did they influence that prediction?
- Which columns are the strongest predictors, which can we ignore?
- How do predictions vary, as we vary these columns?

The nature of a random forest model allows us to answer these questions in a relatively interpretable way.

4.3.1 Tree variance for prediction confidence

One simple way to ascertain the confidence of a model's estimate for any particular year is to use the standard deviation of predictions across the trees, instead of just the mean. This tells us the relative confidence of predictions. In general, we would want to be more cautious of using the results for rows where trees give very different results (higher standard deviations), compared to cases where they are more consistent (lower standard deviations).

The standard deviations from the first five predictions are as follows:

1. 326088.55615766

2. 288820.29037381
3. 97607.73594076
4. 287088.36548729
5. 317326.73513624

Most of these standard deviations are relatively uniform, with the exception of one of the years. This is information that would be useful in a production setting; for instance, if you were using this model to decide on bushfire protection measures for a certain year, a low-confidence prediction might cause you to look more carefully at a year's rainfall data before you put these measures in place.

A histogram of the frequency of certain standard deviations is shown in Figure 4.

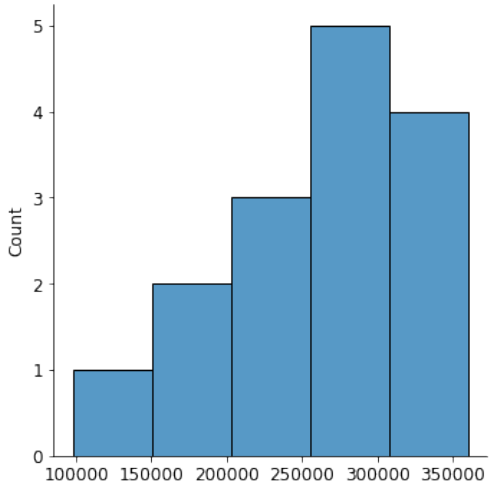

Figure 4: Histogram of tree prediction variance.

4.3.2 Feature importance

We also wish to know what the most important factors are for prediction. We can do this by examining feature importance. To calculate feature importance, we loop through each predicting tree, and at each branch recursively examine what feature was used for that split. We can then determine how much the model improved as a result of that split. The improvement is weighted by the number of rows in that group and added to the importance score for the feature chosen in the split. Finally, the scores are normalised so they add to 1 [3].

Figure 5 shows that the Dennison index for October-January, and the rainfall in January, are the most important predictors of bushfire severity. This differs slightly to our most important splits for the decision tree model above.

4.3.3 Partial dependence

As we've seen, the two most important predictors are the Dennison index (Oct-Jan) and rainfall in January alone. We'd like to understand the relationship between these predictors and bushfire severity.

Partial dependence plots (PDP) show the dependence between the target response and a set of input features of interest, marginalising over the values of all other input features (the *complement* features). Intuitively, we can interpret the partial dependence as the expected target response as a function of the input features of interest [4].

To answer this question, we can't just take the average HA burnt for each Dennison index. The problem with that approach is that many other things vary from year to year as well, such as rainfall in other months. So, merely averaging over all the years that have the same Dennison index would also capture the effect of how every other field also changed along with the Dennison index and how that overall change affected bushfire severity. Instead, what we do is replace every single value in the Dennison index column with 0, and then calculate the predicted bushfire severity for every year, and take the average over all these predictions. Then we do the same for until the highest value of the Dennison index (the year with the most rainfall from October to January). This isolates the effect of only the Dennison index [5].

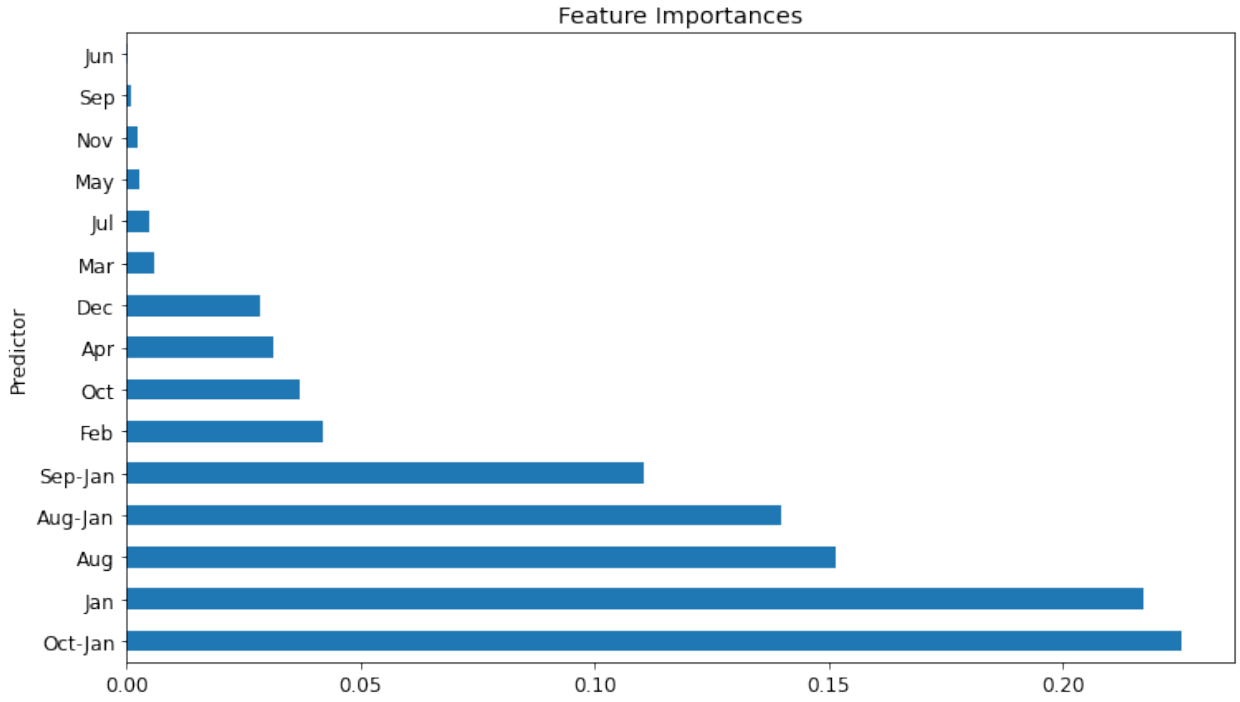

Figure 5: Feature importances for the random forest model.

With these averages, we can then plot each of these years on the x-axis, and each of the predictions on the y-axis. These partial dependence plots are shown in Figure 6.

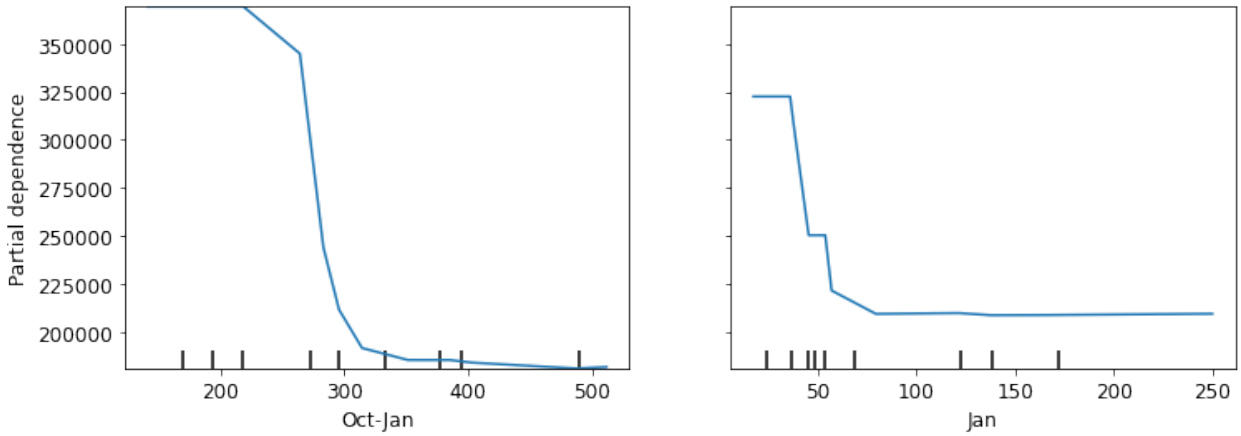

Figure 6: Partial dependence plots for Dennison Index (Oct-Jan) and rainfall in January.

Looking first of all at the Dennison index plot, we see a very interesting relationship. This plot demonstrates that, below a certain threshold of rainfall in these months, bushfire severity is extremely large. However, as soon as the rainfall exceeds this cumulative threshold of approximately 300 mm, bushfire severity drops drastically. This is what we would roughly expect, although the clarity of the drop is startling. A similar relationship holds for rainfall in January alone.

4.3.4 Tree interpreter

We have already seen how to compute feature importances across the entire random forest. The basic idea was to look at the contribution of each variable to improving the model, at each branch of every tree, and then add up all of these contributions per variable.

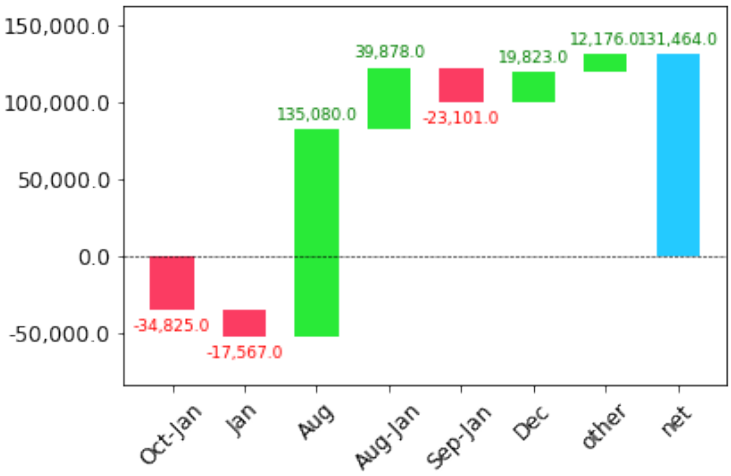

Figure 7: Waterfall plot of feature contributions to prediction for specific year of data.

We can do exactly the same thing, but for just a single row of data. For instance, let's say we are looking at some particular year of data. Our model might predict that this year will have extreme bushfires, and we want to know why. So, we take that one row of data and put it through the first decision tree, looking to see what split is used at each point throughout the tree. For each split, we see what the increase or decrease in the addition is, compared to the parent node of the tree. We do this for every tree, and add up the total change in importance by split variable.

The clearest way to display the contributions is with a waterfall plot. This shows how the positive and negative contributions from all the independent variables sum up to create the final prediction, which is the right-most column labeled "net" in Figure 7.

4.4 Neural network

An initial recurrent neural network with two hidden layers (size 500 and 250 respectively) achieved a validation r-MSE of 1495574 after training for 100 epochs. This is somewhat higher than the r-MSE achieved by the decision tree and random forest above. After removing the outlier of 2019-20, the r-MSE dropped to 111941; this is indeed lower than the previous models. Here, we begin to recognise that our models are balancing competing interests: minimising r-MSE in years when bushfire is relatively regular, periodic and homogeneous, and minimising r-MSE in years with extreme fires.

5 The kitchen sink: classical time-series forecasting methods

5.1 Autoregression

An autoregression model makes an assumption that the observations at previous time steps are useful in predicting the dependent variable at the next time step [6]. This relationship between variables is called correlation. We can use statistical measures to calculate the correlation between the output variable and values at previous time steps at various different lags. The stronger the correlation between the output variable and a specific lagged variable, the more weight the autoregression model will put on that variable when modeling.

There is a quick, visual check that we can do to see if there is an autocorrelation in our time series dataset. We can plot the observation at the previous time step $(t-1)$ with the observation at the next time step $(t+1)$ as a scatter plot. This is shown in Figure 9. Clearly, there is minimal correlation between the previous year's fires and current bushfires. However, this process could be repeated for any other lagged observation, such as if we wanted to review the relationship with the last 10 years of fires.

5.1.1 Autocorrelation plots

We can plot the correlation coefficient for each lag variable. This can very quickly give an idea of which lag variables may be good candidates for use in a predictive model and how the relationship between the observation and its historic values changes over time. The initial autocorrelation plot is shown in Figure 10.

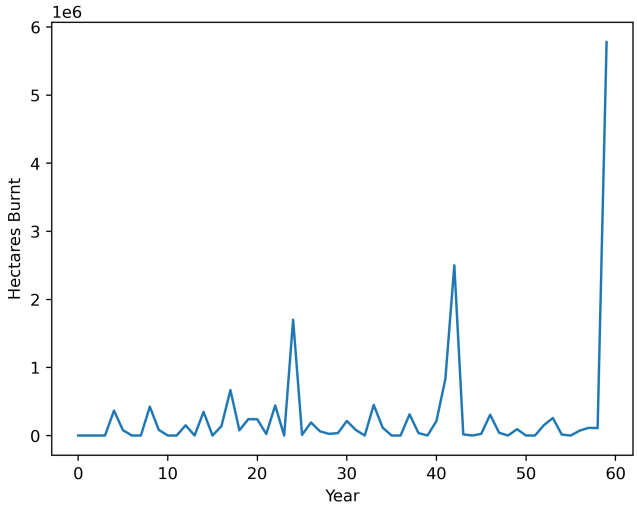

Figure 8: Time-series plot of bushfire severity.

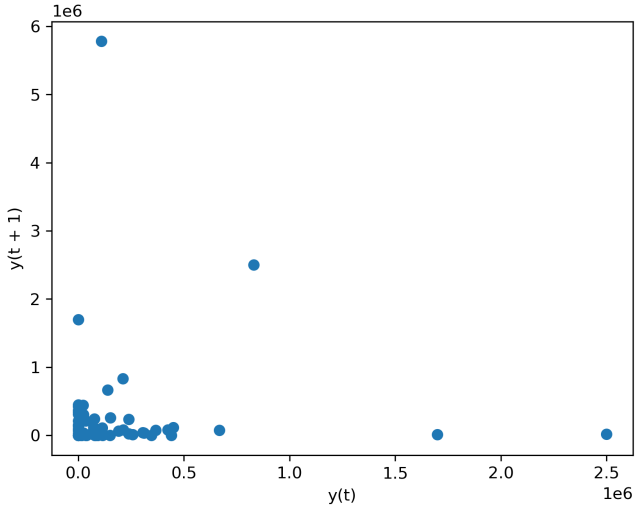

Figure 9: Scatter plot of correlations between previous and current time-step.

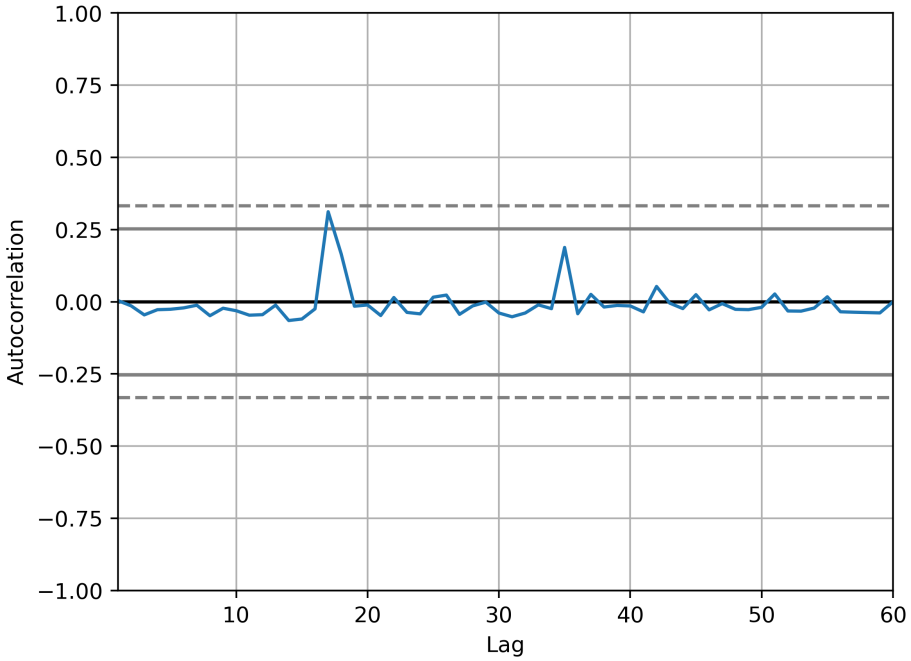

Figure 10: Initial autocorrelation plot. Note the spike in correlation when the lag is 15 periods.

The plot provides the lag number along the x-axis and the correlation coefficient value between -1 and 1 on the y-axis. The plot also includes solid and dashed lines that indicate the 95% and 99% confidence interval for the correlation values. Correlation values above these lines are more significant than those below the line, providing a threshold or cutoff for selecting more relevant lag values. This is shown in Figure 11.

Although most lag variables are uncorrelated with next period's bushfire severity, it is extremely interesting to note the significance of the autocorrelation between the bushfire severity 17 periods ago, and the next period's bushfire. Our autocorrelation analysis has picked up on the spikes in bushfires every 15-20 years, as shown in the initial plot of the data above.

5.1.2 Persistence model

The simplest model that we could use to make predictions would be to persist the last observation. This persistence model provides a baseline of performance for the problem that we can use for comparison with an autoregression model. We can develop a test harness for the problem by splitting the observations into training and test sets, with only the last 15 years in the dataset assigned to the test set as "unseen" data that we wish to predict.

The predictions are made using a walk-forward validation model so that we can persist the most recent observations for the next day. This means that we are not making a 15-year forecast, but 15 1-year forecasts. This base model is shown in Figure 12. The r-MSE for this model is incredibly high: 17977600000.

5.1.3 Autoregression model

This model has an r-MSE of 369585, which is comparable to the tree-based models above. This immediately tells us that past bushfire patterns are at least as good at predicting future bushfires as current rainfall. The model's predictions are shown in Figure 13.[2]

[2]Note that there is nothing to prevent the model from predicting negative values. In that case, we clamp any negative predictions at 0. The random forest models did not have this problem, as they work by averaging predictions across observations, and thus cannot predict outside the range of observations. Regression models do not have this feature.

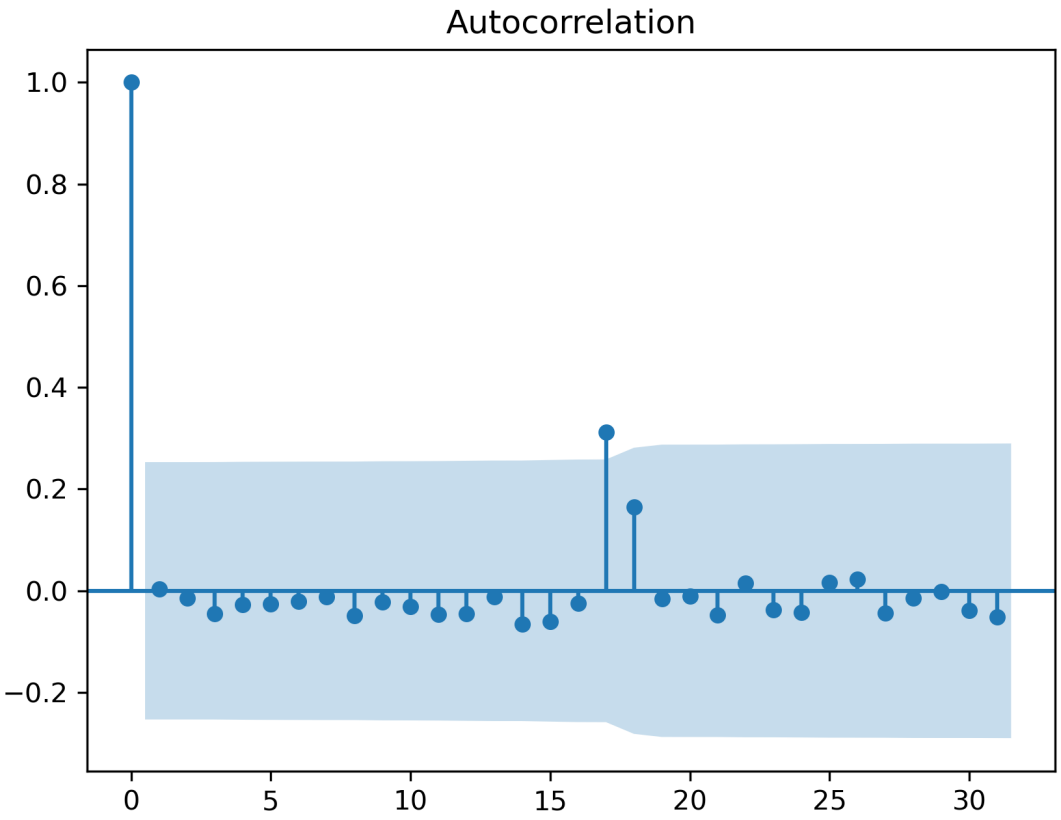

Figure 11: Autocorrelation plot showing spike approximately every 15 years.

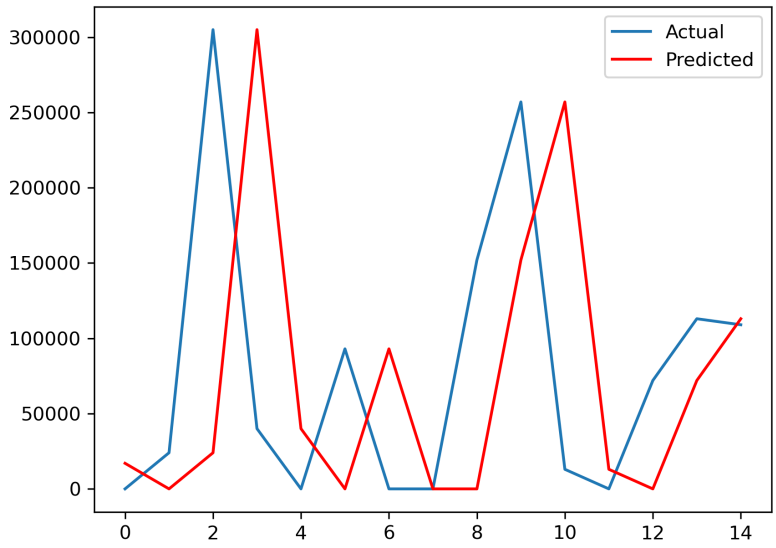

Figure 12: Persistence model which simply uses the last observation as a prediction.

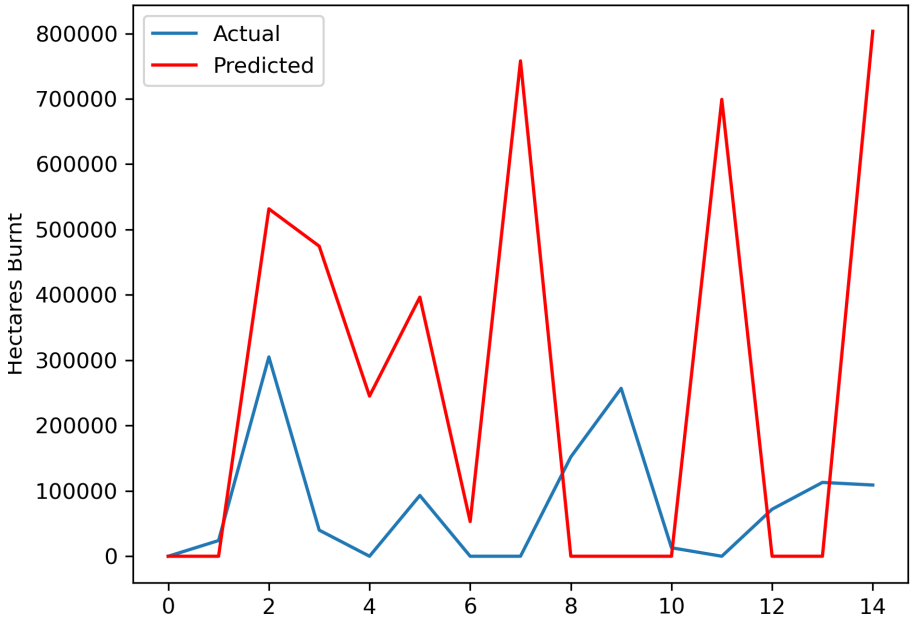

Figure 13: Initial autoregression model - seven one-year forecasts.

Our model does not make it easy to update the model as new observations become available. One way would be to re-train the autoregression model each year as new observations become available. An alternative would be to use the learned coefficients and manually make predictions. This requires that the history of 15 prior observations be kept and that the coefficients be retrieved from the model and used in the regression equation to generate new forecasts.

The coefficients are provided in an array with the intercept term followed by the coefficients for each lag variable starting at $t-1$ to $t-n$. We simply need to use them in the right order on the history of observations. This new model achieves a r-MSE of 305602 HA, which is lower than the previous AR model.

5.2 Moving average

5.2.1 Simple moving average

The *simple moving average* is the unweighted mean of the previous M data points. The selection of M (sliding window) depends on the amount of smoothing desired. Increasing the value of M improves the smoothing at the expense of accuracy. For a sequence of values, we calculate the simple moving average at time period t as follows:

$$SMA_t = \frac{x_t + x_{t-1} + ... + x_{M-(t-1)}}{M}$$

The results of a simple moving average is shown in Figure 15.

5.2.2 Cumulative moving average

The *cumulative moving average* is the unweighted mean of the previous values up to the current time t. The simple moving average has a sliding window of constant size M. On the contrary, the window size becomes larger as the time passes when computing the cumulative moving average. The CMA is shown in Figure 16. Interestingly, the cumulative moving average makes it clear that average hectares burnt is steadily increasing over time, with sharp jumps in average every 15-20 years.

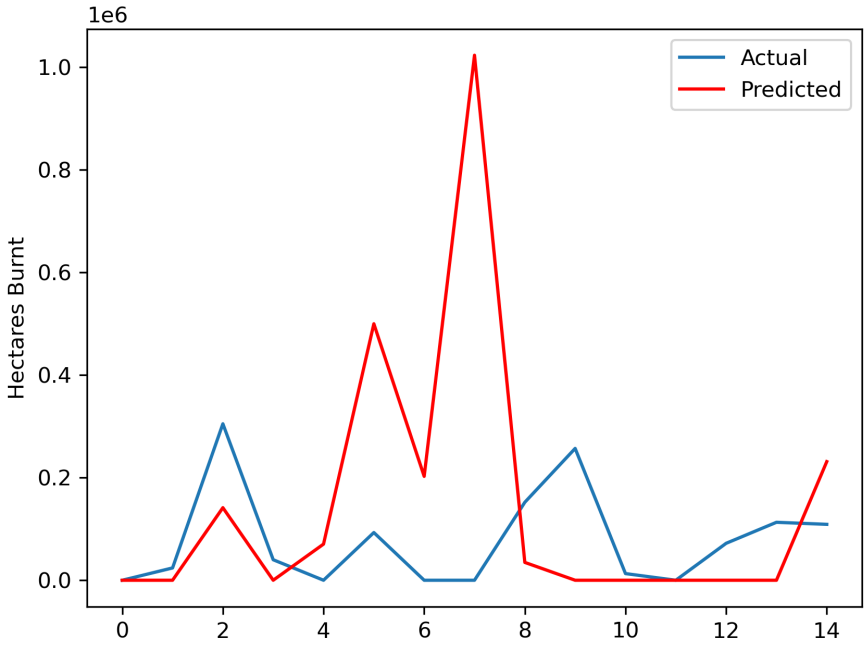

Figure 14: Second auto-regression model using learned coefficients to make predictions.

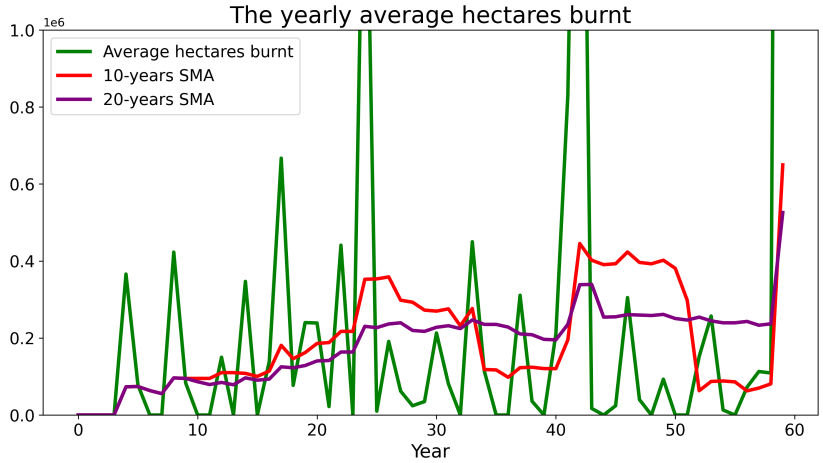

Figure 15: Simple moving average of bushfire data.

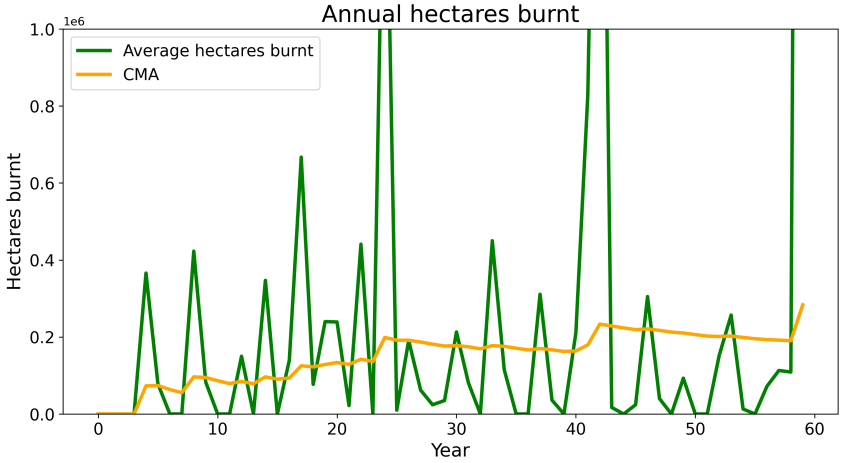

Figure 16: Cumulative moving average.

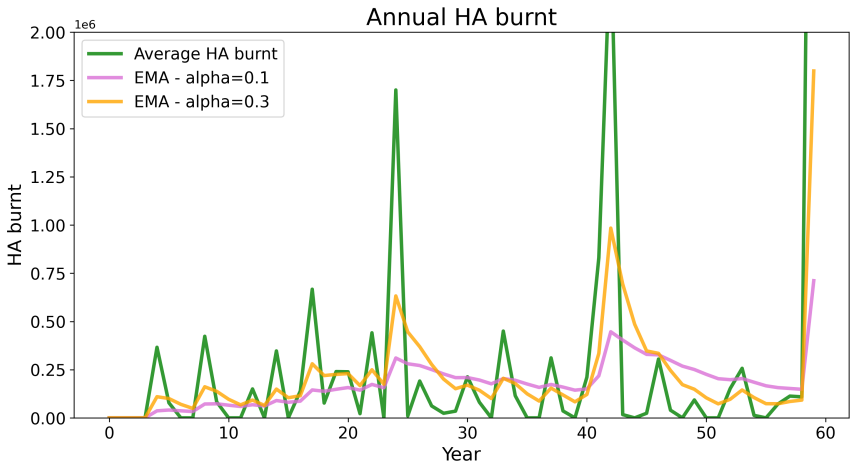

Figure 17: Exponential moving average.

5.2.3 Exponential moving average

The exponential moving average is a widely used method to filter out noise and identify trends. The weight of each element decreases progressively over time, meaning the exponential moving average gives greater weight to recent data points. This is done under the idea that recent data is more relevant than old data.

The algebraic formula to calculate the exponential moving average at the time period t is:

$$EMA_t = \begin{cases} x_0 & t = 0 \\ \alpha x_t + (1-\alpha) EMA_{t-1} & t > 0 \end{cases}$$

Here, α is the smoothing factor, where $\alpha \in [0, 1]$. Two different smoothing factors of 0.1 and 0.3 are shown in Figure 17. This plot shows even more clearly the jump in moving average every 15-20 years.

5.3 Autoregressive integrated moving average (ARIMA)

An ARIMA model explicitly caters to a suite of standard structures in time series data, and as such provides a simple yet powerful method for making skillful time series forecasts. ARIMA is an acronym that stands for AutoRegressive Integrated Moving Average. It is a generalisation of the simpler autoregressive Moving Average and adds the notion of integration.

Each of these components are explicitly specified in the model as a parameter. A standard notation is used of ARIMA(p, d, q). The parameters of the ARIMA model are defined as follows:

- p: The number of lag observations included in the model, also called the lag order.
- d: The number of times that the raw observations are differenced, also called the degree of differencing.
- q: The size of the moving average window, also called the order of moving average.

A linear regression model is constructed including the specified number and type of terms, and the data is prepared by a degree of differencing in order to make it stationary, i.e. to remove trend and seasonal structures that negatively affect the regression model. First, we fit an ARIMA model to the entire time-series and analyse the residual errors. Figure 18 shows that the errors are Gaussian, but may not be centered on zero. This is confirmed by a non-negative mean in the residuals.

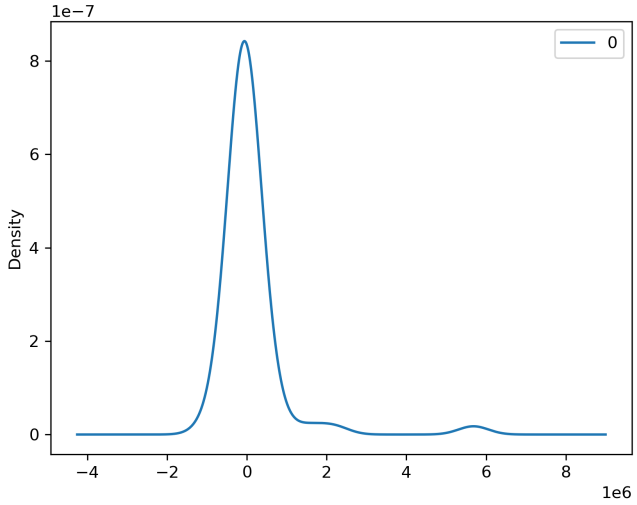

Figure 18: Line plot of residuals.

Doing a grid search on the parameters for the ARIMA model, we find that the lowest r-MSE actually comes from values of $(p, q, d) = (16, 1, 5)$. We get an r-MSE of 368397. The forecasts are shown in Figure 19.

5.4 Seasonal Autoregressive Integrated Moving-Average (SARIMA)

A problem with ARIMA is that it does not support seasonal data. ARIMA expects data that is either not seasonal or has the seasonal component removed, e.g. seasonally adjusted via methods such as seasonal differencing.

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component. It adds three new hyperparameters to specify the autoregression (AR), differencing (I) and moving average (MA) for the seasonal component of the series, as well as an additional parameter for the period of the seasonality.

There are additional hyperparameters for a SARIMA model, in addition to the ARIMA hyperparameters above:

- P: Seasonal autoregressive order.
- D: Seasonal difference order.
- Q: Seasonal moving average order.
- m: The number of time steps for a single seasonal period.

Because SARIMA models are more computationally expensive, grid searches are not feasible in a short time period. The model chosen, partly using a guess-and-check procedure, uses values of $P = 1, D = 0, Q = 1, m = 17$, where $m = 17$ was chosen because the seasonal component appears every 15-20 years. This model gave the lowest r-MSE achieved by a non-RNN model yet, with a value of 175593 hectares. The model forecast is shown in Figure 20.

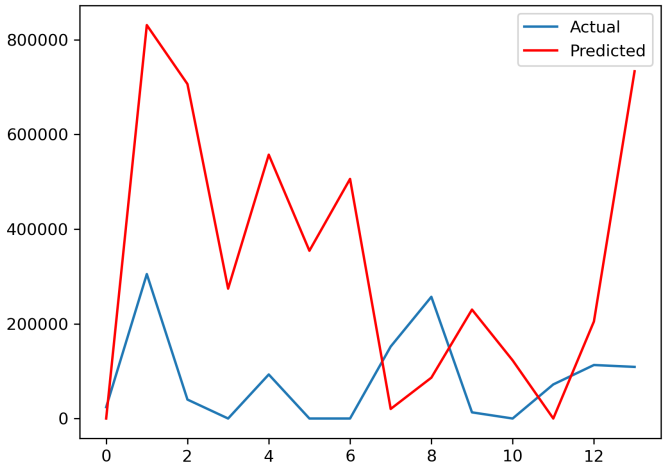

Figure 19: ARIMA model.

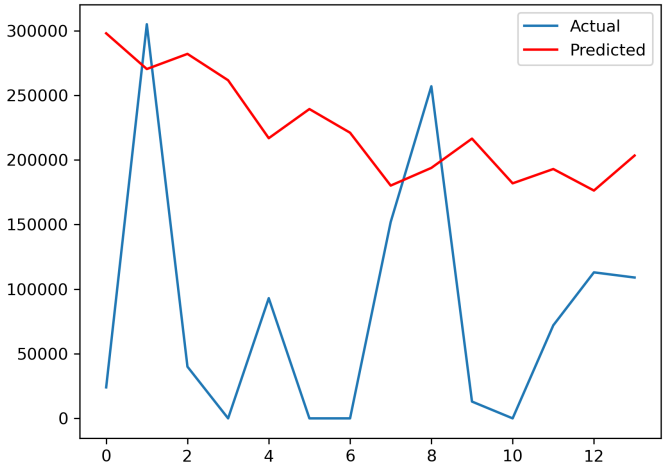

Figure 20: Adding a seasonal component to the ARIMA model.

5.5 Fourier extrapolation

The Fourier transform gives us a method of decomposing periodic functions into their sinusoidal components. That is, we attempt to gain information above the frequency and amplitude of components of the time series, specifically the peak in bushfires every 15-20 years.

We begin by applying the Fourier transformation to the time series, mapping it to the Fourier space with frequency on the x-axis, and amplitude on the y-axis. This is shown in Figure 21.

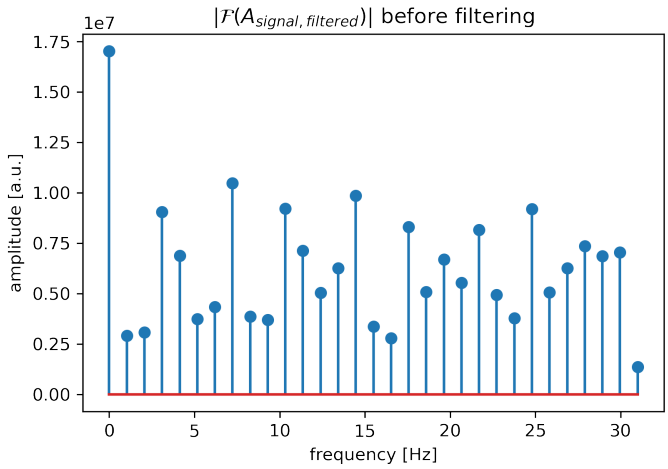

Figure 21: The bushfire time-series in the Fourier space.

We then apply a filter where we zero all elements of the Fourier series below a certain amplitude, which will remove some of the noise from the time-series. That is, this filter will smooth the data such that sinusoidal movements are more evident.

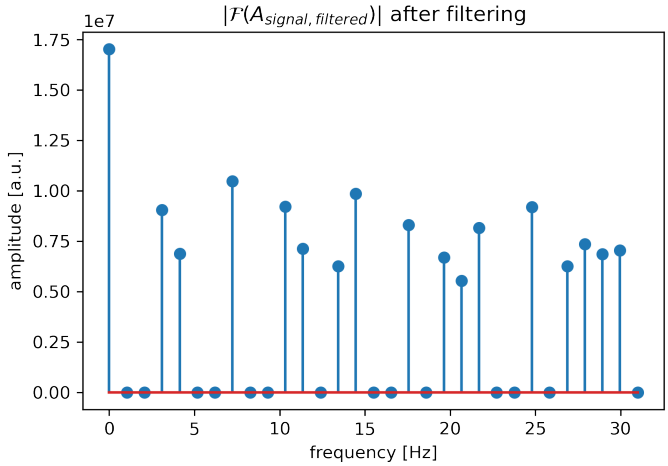

Figure 22: Filtering only those elements with an amplitude above a chosen threshold.

We then apply the inverse Fourier transform and compare it to the original series. Interestingly, our filter has not really denoised the data - there are smaller waves in between the larger spikes. This seems to suggests a bi-annual pattern of bushfire peaking, in addition to (albeit smaller than) the peaks every 15 years.

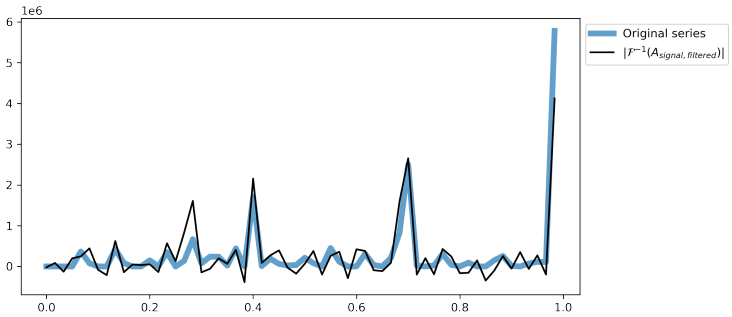

Figure 23: The filtered Fourier series inverted back to the original space. This has denoised the data somewhat, and gives us coefficients for the frequency of large bushfire events, allowing for extrapolation.

We can then extrapolate this Fourier series into the future. Noting that bushfire peaks seem to get larger with each cycle, we manually increase amplitude by adjusting the Fourier coefficients as we move along the x-axis. This suggests that another severe bushfire will come after the 80th period (2039).

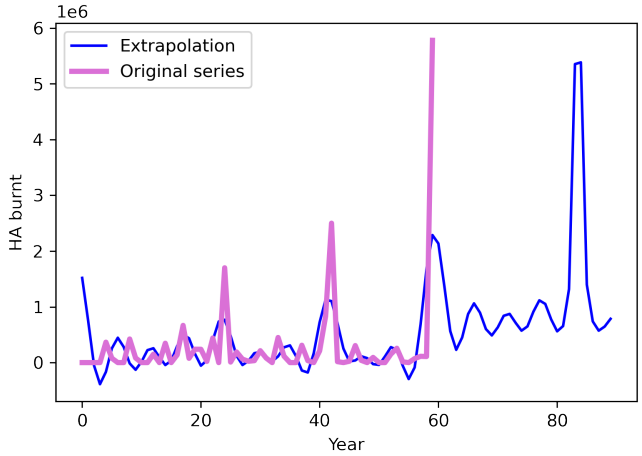

Figure 24: Fourier extrapolation for the time series. Note the upward trend.

6 Fully-focused feature engineering

Combining all of the above, we want to extract as much data as possible from the time-series through the process of feature-engineering. As seen above, models that focus on extracting this information themselves do not perform well on our time-series, as minimising the loss forces the forecast into a constant. We want a form of model that explicitly knows when large jumps in bushfire severity are coming.

Consequently, we discard all rainfall data except the three Dennison indices, and add rolling measures of data that aim to capture past patterns in bushfire severity. The additional predictor variables are:

- *Exponential moving average of hectares burnt:* here we use a smoothing factor of $\alpha = 0.1$, to ensure that long-term trends (15+ years) are captured.
- *Simple moving average of hectares burnt:* this is an unweighted average of the last 15 years of HA burnt.
- *Exponential moving standard deviation of hectares burnt:* again we use a smoothing factor of $\alpha = 0.1$, to ensure that long-term trends (15+ years) are captured.
- *Sum of hectares burnt:* we sum HA burnt over the last 15 years.
- *Years since million HA burnt:* this variable aims to capture the big spikes which occur every 15-20 years.

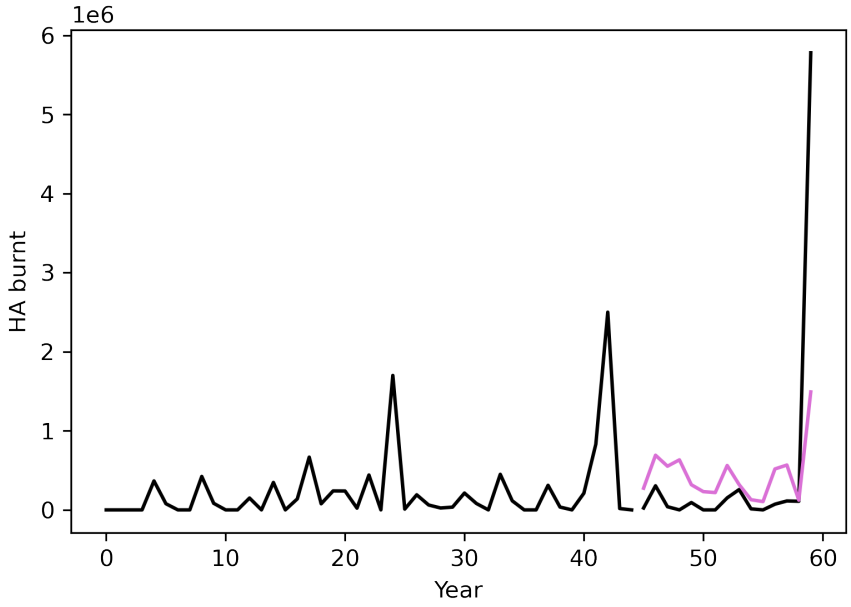

Figure 25: Random forest model built with heavy feature-engineering.

- *Exponential moving average of Dennison index:* here we use a smoothing factor of $\alpha = 0.4$, to ensure that the focus is on recent rainfall.
- *Simple moving average of Dennison index:* this is an unweighted average of the last 15 years of rainfall in October-January.
- *Exponential moving standard deviation of Dennison index:* again we use a smoothing factor of $\alpha = 0.4$, to ensure that the focus is on recent rainfall.
- *Sum of Dennison index:* we sum HA burnt over the last 15 years.

6.1 Random forest

We again fit a random forest regressor model to this data. Again, the first 45 years are used for training the model, and the last 15 years are used to test it. Whilst this gives an r-MSE of 1155211 hectares, this is the lowest out of any model when including the considerable outlier of the 2019-20 bushfires. Promisingly, this model picks up on the fact that there should be a significant jump in 2019-20. However, yet again the model fails to account for the severity of the jump.

The importance rankings of these variables are shown in Table 5. Clearly, the most important parts of the model are the past patterns in bushfire severity. Rainfall, whilst it does feature, is significantly less important.

Table 5: Importance measures for predictor variables in feature-engineered dataset.

Feature	Importance
Exponential moving standard deviation (HA burnt)	0.17
EMA (HA burnt)	0.15
Years since million HA burnt	0.13
SMA (HA burnt)	0.14
Dennison index (Oct-Jan)	0.11
SMA Dennison index (Oct-Jan)	0.07
Sum HA burnt (Last 15 years)	0.06
EMA Dennison index (Oct-Jan)	0.05

Plotting the partial dependence for years since million HA burnt, we see a clear, quasi-exponential increase of bushfire risk with each additional year since a major fire. This is shown in Figure 26.

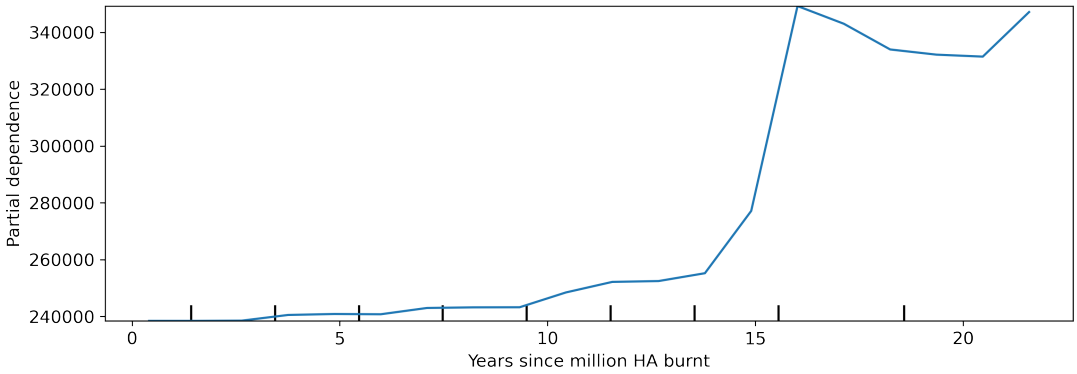

Figure 26: Partial dependence of HA burnt on years since a million HA burnt.

Interestingly, we can also generate a multivariable plot of the partial dependence of HA burnt conditioned on both the Dennison index and years since a million HA were burnt. Evidently, bushfire severity is highest in the back upper left quadrant, where there has been little rainfall and 10+ years since a severe bushfire. Whilst intuitive, it is reassuring to see the model reflect this in Figure 27.

6.1.1 Sensitivity Analysis

Sensitivity analysis is the study of how uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input factors [7]. The first order sensitivity index, which measures the additive linear effect of the variable on the output, is written as:

$$S_i = \frac{V_{X_{\tilde{i}}}(E_{X_{\tilde{i}}}Y|X_i))}{V(Y)}$$

Similarly, the total sensitivity index measures the total effect, i.e. first and higher order effects (interactions) of factor X_i. It is given by

$$\begin{aligned} S_{Ti} &= \frac{E_{X_{\tilde{i}}}(V_{X_i}(Y|X_{\tilde{i}}))}{V(Y)} \\ &= 1 - \frac{V_{X_{\tilde{i}}}(E_{X_i}(Y|X_{\tilde{i}}))}{V(Y)} \end{aligned}$$

Here, we use Sobol sampling. The Sobol sequence is a popular quasi-random low-discrepancy sequence used to generate uniform samples of parameter space [8].

The variables with the highest first-order sensitivity effects were years since one million HA were burnt, as well as the moving average of the standard deviation of HA burnt. These predictors had first-order sensitivity indices of 0.51 and 0.20 respectively. The highest total sensitivity indices out of predictor variables are shown in Table 6. Evidently, our results from above are reflected here. Both of the important first-order variables remain the most important parts of the model by far, when measured by total sensitivity indices.

The fact that the first-order sensitivity indices closely match the total sensitivity indices suggests a lack of higher-order interaction between predictor variables (at least the important ones).

6.2 LSTM

Long short-term memory networks are a form of recurrent neural network (RNN) which differ in their implementation of feedback connections. Although RNNs can maintain previous information and use it for a current predictions task, they struggle to learn when there is a significant temporal distance between the relevant information and the current prediction. These long-term dependencies, such as periodic bushfire spikes, thus require a certain subset of RNNs: LSTMs.

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer. However, in LSTMs this structure

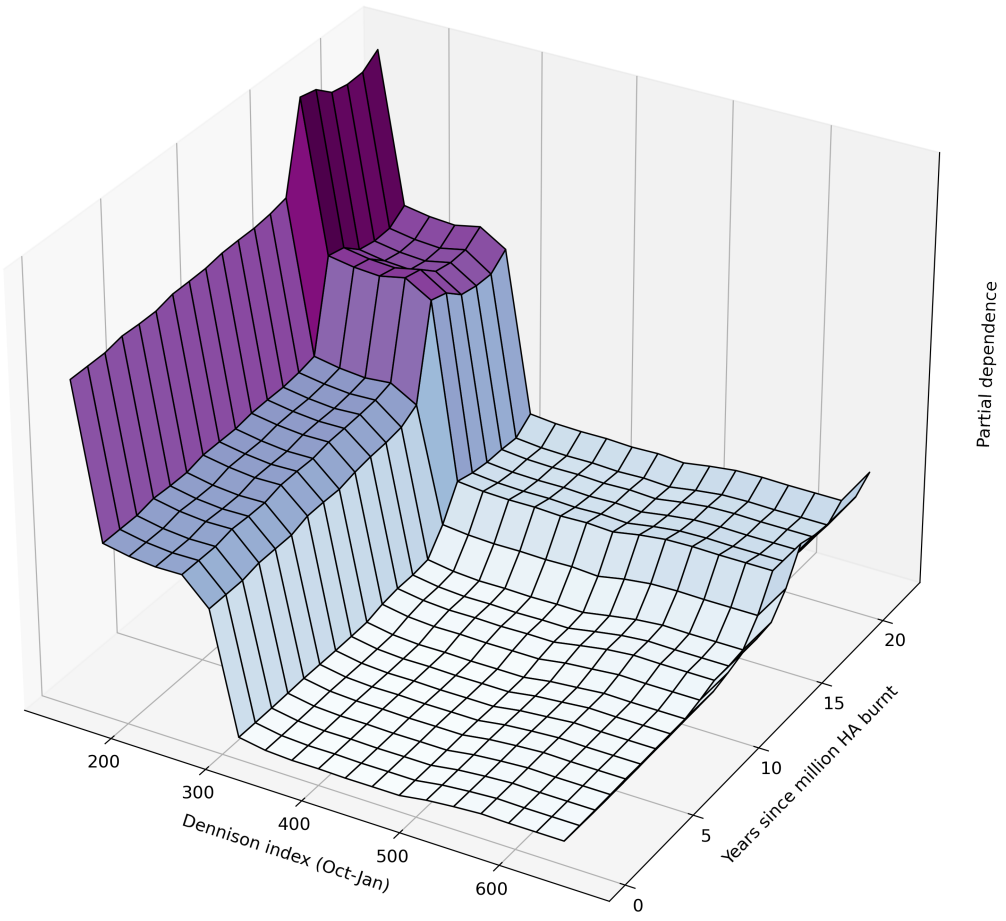

Figure 27: Multivariable partial dependence plot of HA burnt, conditioned on both the Dennison index and years since a million HA were burnt.

Table 6: Total sensitivity indices for predictor variables.

	Total sensitivity index	ST_conf
Years since million HA burnt	0.532793	0.054314
Exponential moving std (HA burnt)	0.205694	0.018832
Dennison index (Oct-Jan)	0.082034	0.010347
EMA Dennison index (Oct-Jan)	0.047667	0.004827
Dennison index (Sep-Jan)	0.034586	0.003668
Sum HA burnt (Last 15 years)	0.030087	0.003177
Dennison index (Aug-Jan)	0.022516	0.002847
EMA (HA burnt)	0.021157	0.004053

is implemented to have four rather than one neural network layer. A common LSTM unit is composed of a cell, an input gate, an output gate, and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell [9].

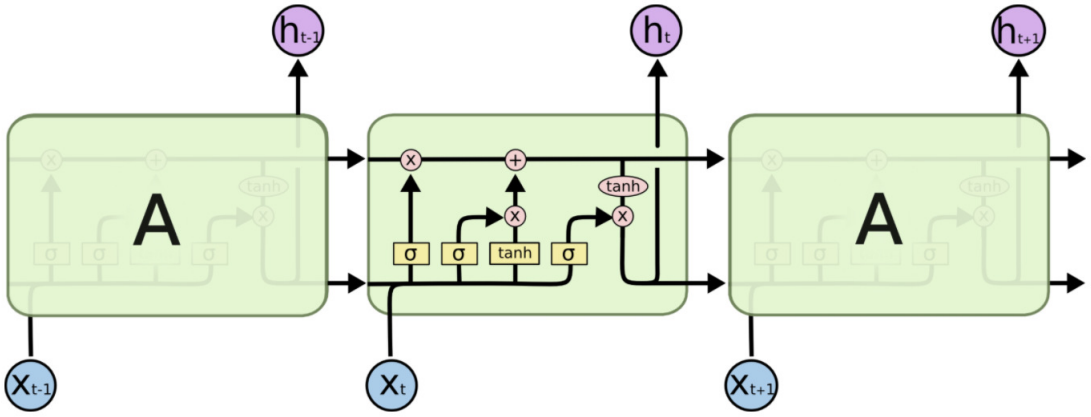

Figure 28: Training process for LSTM model

The LSTM learning process can be summarised as follows:

1. The forget gate f_t determines the level of information retained in the cell state, considering the output from the previous cell h_{t-1} and new information x_t:
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
2. An input gate i_t determines which values in the cell state to update:
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
3. A tanh layer creates a vector of new candidate values, C_t, which can be added to the cell state:
$$i_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$
4. The cell state is updated by multiplying the previous state by f_t, and adding the new candidate values:
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \ C_t$$
5. A sigmoid layer determines which parts of the cell state to output, and then the cell state is put through $\tanh$ to scale values to between -1 and 1, and multiply by the sigmoid layer to output only the relevant parts:
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t \times \tanh C_t$$

LSTMs are well-suited to regression tasks on time series data, particularly when the duration between data points varies. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods.

To begin, the data was normalised. The models were designed to make a set of predictions based on a window of consecutive samples from the data. The windows were constructed from the width of the inputs and labels (represented in number of time-steps). The RNN using LSTM units was trained on sets of bushfire annual data. The training was undertaken using an optimisation algorithm of gradient descent with backpropagation through time. This achieved a best 864976 r-MSE on the full validation set. As shown below, the model clearly picks up on the spike in 2019-20, albeit as a smaller spike than the actual value (Figure 29).

7 Conclusion and comparison of models

Initial implementations of regression models confirmed the non-linearity of the relationship between rainfall and bushfire data, as well as the inherent limitations in in using rainfall as a predictor. However, tree-based methods greatly

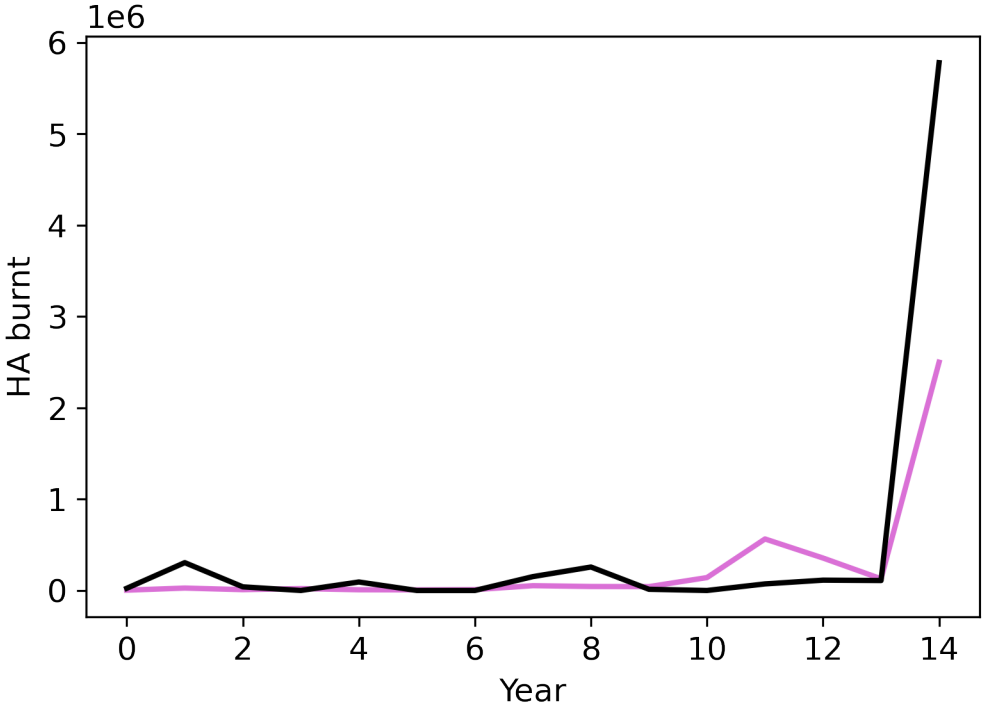

Figure 29: Predictions from the LSTM, showing the ability to predict the spike in 2019-20.

improved on regression r-MSE, and confirmed that the Dennison index (Oct-Jan) is important in determining fires in any particular year.

Classical time-series forecasting methods were overall disappointing in capturing inherent trends in the data. This is due to the highly non-monotonic and non-linear nature of the time-series. Even the seasonality component is non-linear, as the peaks themselves grow with each cycle through the harmonics. This renders models such as autoregression and SARIMA largely ineffective for forecasting. However, these classical methods gave us ideas for the appropriate features to engineer for future models, which resulted in greater success for bootstrapped machine learning models such as random forests. Additionally, SARIMA ended up achieving similar results to some tree-based models, and even the RNN, when the final 2019-20 outlier was removed.

One disappointing aspect of these models was their inability to capture how the outlying bushfires, which occur every 15-20 years, are seemingly getting larger. However, this is to be expected; if the models are only trained on the first 45 years of data, then there are only two of these data points to learn from, which makes it difficult to extrapolate any pattern. We saw greater success when manually adjusting the sine and cosine coefficients in the Fourier series approximation of the time series, but a future direction would involve building this increasing amplitude of specific spikes into the model itself.

The final random forest and LSTM models based on a heavily feature-engineered dataset provided the most insight into the data. Clearly, it is previous bushfire severity that is the largest determinant of bushfires in the current period, although rainfall does factor in somewhat. 3D partial dependence plots based on this model revealed a highly intuitive relationship between rainfall, years since a severe fire, and HA burnt: low rainfall and a lengthy period since a million HA burnt (15+ years) give the greatest risk of a significant fire in the current period. This conclusion was bolstered by a Sobol sensitivity analysis of the relevant variables.

The performance of all models is summarised in Table 7. The bolded entries in each column highlight the best performing model. In both cases, the best model was driven by deep learning, rather than classical time-series forecasts or machine learning.

Table 7: Performance (r-MSE) summary of all models.

Model	r-MSE (Full validation set)	r-MSE (2019-20 outlier removed)
Linear regression	1472722	323087
Decision tree	1472661	177026
Random forest (raw data)	1401525	285326
RNN	1495574	**111941**
Autoregression	1086393	305602
ARIMA	1362411	368397
SARIMA	1478002	175593
Random forest (feature-engineered)	1155211	341027
LSTM (feature-engineered)	**864976**	368875

8 Next steps

- **Vector autoregression:** using multiple inputs at each step of the autoregressive model. This would allow us to incorporate rainfall data, not just the univariate bushfire time-series.
- **Locally stationary wavelets:** a disadvantage of the Fourier extrapolation is that it simply repeats the cycle of harmonics slanted on the linear trend. Instead, we could determine how the frequencies change as time passes.
- **Modelling the increasing amplitude of certain spikes in the Fourier time series:** whilst we manually adjusted the Fourier coefficients by multiplying certain frequencies, future work could see this increasing amplitude built into the Fourier series approximation itself.
- **Generate synthetic extensions of the data to force model to learn spikes:** although unorthodox, generating an extension of the pattern and feeding this in as training data may allow models to recognise the peaks that occur at regular intervals, and the fact that these peaks are exponentially increasing.
- **Classification model to predict severe bushfires:** most of the models above try to balance opposing considerations: ensuring r-MSE is low in years where there are little to no bushfires, and accounting for years where the fires are significantly large. An alternative to a regression model would be to train a classification model to predict years where there is a fire of over 1,000,000 HA burnt.

References

[1] Mr Brijain, R Patel, MR Kushik, and K Rana. A survey on decision tree algorithm for classification. 2014.

[2] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[3] Yanjun Qi. Random forest for bioinformatics. In *Ensemble machine learning*, pages 307–323. Springer, 2012.

[4] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[5] Jeremy Howard and Sylvain Gugger. Fastai: a layered api for deep learning. *Information*, 11(2):108, 2020.

[6] Hirotugu Akaike. Autoregressive model fitting for control. In *Selected Papers of Hirotugu Akaike*, pages 153–170. Springer, 1998.

[7] Andrea Saltelli, Paola Annoni, Ivano Azzini, Francesca Campolongo, Marco Ratto, and Stefano Tarantola. Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. *Computer physics communications*, 181(2):259–270, 2010.

[8] Ilya M Sobol. Sensitivity analysis for non-linear mathematical models. *Mathematical modelling and computational experiment*, 1:407–414, 1993.

[9] Felix A Gers, Nicol N Schraudolph, and Jürgen Schmidhuber. Learning precise timing with lstm recurrent networks. *Journal of machine learning research*, 3(Aug):115–143, 2002.