Modelling the transition to a low-carbon energy supply

Alexander John Michael Kell

School of Computing
Newcastle University

This dissertation is submitted for the degree of
Doctor of Philosophy

July 2020

I would like to dedicate this thesis to my family and my loving parents…

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Alexander John Michael Kell
July 2020

Acknowledgements

And I would like to acknowledge ...

Abstract

This is where you write your abstract ...

Table of contents

List of figures

List of tables

Glossary

Agent Based Model A simulation technique made up of individual agents. 1.1

Artificial Neural Network A machine learning algorithm that is modelled on the brain and made up of multiple layers of neurons and weights. 1.1, 5.3

Support Vector Regression A machine learning algorithm used for regression which depends only on a subset of the training data. 1.1

Acronyms

ABM Agent Based Model. 4.2, 4.4, 4.6, 4.11

SVR Support Vector Regression. 5.2, 5.3, 5.6, 5.9, 5.12

Chapter 1

Introduction

1.1 What is loren ipsum? Title with math σ

Agent Based Model goes here.

Artificial Neural Network goes here.

Support Vector Regression goes here.

Chapter 2

Background

2.1 Reasonably long section title

Test words

Chapter 3

Literature review

Literature review goes here.

Chapter 4

ElecSim model

4.1 e-Energy 2019 Paper

4.2 Introduction

The world faces significant challenges from climate change [120]. A rise in carbon emissions increases the risk of severe impacts on the world such as rising sea levels, heat waves and tropical cyclones [120]. A survey [34] showed that 97% of scientific literature concurs that the recent change in climate is anthropogenic.

High carbon emitting electricity generation sources such as coal and natural gas currently produce 65% of global electricity, whereas low-carbon sources such as wind, solar, hydro and nuclear provide 35% [20]. Hence, to bring about change and reach carbon-neutrality, a transition in the electricity mix is required.

Due to the long construction times, operating periods and high costs of power plants, investment decisions can have long term impacts on future electricity supply [26]. Governments and society, therefore, have a role in ensuring that the negative externalities of emissions are priced into electricity generation. This is most likely to be achieved via carbon tax and regulation to influence electricity market players such as generation companies (GenCos).

Decisions made in an electricity markets may have unintended consequences due to their complexity. A method to test hypothesis before they are implemented would therefore be useful.

Simulation is often used to increase understanding as well as to reduce risk and reduce uncertainty. Simulation allows practitioners to realise a physical system in a virtual model. In this context, a model is defined as an approximation of a system through the use of mathematical formulas and algorithms. Through simulation, it is possible to test a system where real life experimentation would not be practical due to reasons such as prohibitively high costs, time constraints or risk of detrimental impacts. This has the dual benefit of minimising the risk of real decisions in the physical system, as well as allowing practitioners to test less risk-averse strategies.

Agent Based Model (ABM)s are a class of computational simulation models composed of autonomous, interacting agents and model the dynamics of a system. Due to the numerous and diverse actors involved in electricity markets, ABMx have been utilised in this field to address phenomena such as market power [137].

This paper presents ElecSim, an open-source ABM that simulates GenCos in a wholesale electricity market. ElecSim models each GenCo as an independent agent and electricity demand. An electricity market facilitates trades between the two.

GenCos make bids for each of their power plants. Their bids are based on the generator's short run marginal cost (SRMC) [130], which excludes capital and fixed costs. The electricity market accepts bids in cost order, also known as merit-order dispatch. GenCos invest in power plants based on expected profitability.

ElecSim is designed to provide quantitative advice to policy makers, allowing them to test policy outcomes under different scenarios. They are able to modify a script to realise a scenario of their choice. It can also be used by energy market developers who can test new electricity sources or policy types, enabling the modelling of changing market conditions.

The contribution of this paper is a new open-source framework with example scenarios of varying carbon taxes. We provide curated data, and improve realism via Monte-Carlo sampling. Section **??** is a literature review. Section **??** details the model and assumptions made, and Section **??** provides performance metrics and validation. Section **??** details our results. We conclude the work in Section **??**.

4.2.1 Literature Review

Live experimentation of physical processes is often not practical. The costs of real life experimentation can be prohibitively high, and can require significant time in order to fully ascertain the long-term trends. There is also a risk that changes can have detrimental impacts and lead to risk-averse behaviour. These factors are true for electricity markets, where decisions can have long term impacts. Simulation, however, can be used for rapidly prototyping ideas. The simulation is parametrised by real world data and phenomena. Through simulation, the user is able to assess the likelihoods of outcomes under certain scenarios and parameters [110].

Energy models can typically be classified as top-down macro-economic models or bottom-up techno-economic models [18]. Top-down models typically focus on behavioural realism with a focus on macro-economic metrics. They are useful for studying economy-wide responses to policies [70], for example MARKAL-MACRO [50] and LEAP [74]. Bottom-up models represent the energy sector in detail, and are written as mathematical programming problems [59].

It is possible to further categorise bottom-up models into optimisation and simulation models. Optimisation energy models minimise costs or maximise welfare, defined as the material and physical well-being of people [100]. Examples of optimisation models are MARKAL/TIMES [50] and MESSAGE [143].

However, electricity market liberalisation in many western democracies has changed the framework conditions. Centralised, monopolistic, decision making entities have given way to multiple heterogeneous agents acting for their own best interest [121]. Policy options must therefore be used to encourage changes to attain a desired outcome. It is proposed that these complex agents are modelled using ABMs due to their non-deterministic nature.

Tool name	Open Source	Long-Term Investment	Market	Stochastic Inputs	Country Generalisability
SEPIA [71]	✓	×	✓	Demand	✓
EMCAS [Conzelmann et al.]	×	✓	✓	Outages	✓
NEMSIM [14]	?	✓	✓	×	×
AMES [149]	✓	×	Day-ahead	×	×
GAPEX [30]	?	×	Day-ahead	×	✓
PowerACE [139]	×	✓	✓	Outages Demand	✓
EMLab [26]	✓	✓	Futures	×	✓
MACSEM [134]	?	×	✓	×	✓
ElecSim	✓	✓	Futures	✓	✓

Table 4.1 Features of electricity market ABM tools.

Traditional centralised optimisation models are not designed to describe a system which is out of equilibrium. Optimisation models assume perfect foresight and risk neutral investments with no regulatory uncertainty. The core dynamics which emerge from equilibrium remain a black-box. For example, the model assumes a target will be reached, and does not provide information for which this is not the case. Reasons for this could be investment cycles which move the model away from equilibrium [26].

A number of ABM tools have emerged over the years to model electricity markets: SEPIA [71], EMCAS [Conzelmann et al.], NEMSIM [14], AMES [149], GAPEX [30], PowerACE [139], EMLab [26] and MACSEM [134]. Table 4.1 shows that these do not suit the needs of an open source, long-term market model. We will demonstrate that Monte-Carlo sampling of parameters is also required to increase realism.

There have been a number of recent studies using ABMs which focus on electricity markets, however they often utilize ad-hoc tools which are designed for a particular application [141, 69, 109]. ElecSim, however, has been built for re-use and reproducibility. The survey [160] cites that many of these tools do not release source code or parameters, which is a problem that ElecSim seeks to address.

Table 4.1 contains six columns: tool name, whether the tool is open source or not, whether they model long-term investment in electricity infrastructure, and the markets they model. We determine how the stochasticity of real life is modelled, and determine whether the model is generalisable to different countries.

An open source toolkit is important for reproducibility, transparency and lowering barriers to entry. It enables users to expand the model to their requirements and respective country. The modelling of long-term investment enables scenarios to emerge, and enable users to model investment behaviour. We demonstrate that the use of a Monte-Carlo method improves results.

SEPIA [71] is a discrete event ABM which utilises Q-learning to model the bids made by GenCos. SEPIA models plants as being always on, and does not have an independent system operator (ISO), which in an electricity market, is an independent non-profit organization for coordinating and controlling of regular operations of the electric power system and market [165]. SEPIA does not model a spot market, instead focusing on bilateral contracts. As opposed to this, ElecSim has been designed with a merit-order, spot market in mind. As shown in Table 4.1, SEPIA does not include a long-term investment mechanism.

EMCAS [Conzelmann et al.] is a closed source ABM. EMCAS investigates the interactions between physical infrastructures and economic behaviour of agents. However, ElecSim focuses on the dynamics of the market, and provides a simplified, transparent model of market operation, whilst maintaining robustness of results.

NEMSIM [67] is an ABM that represents Australia's National Electricity Market (NEM). Participants are able to grow and change over time using learning algorithms. NEMSIM is non-generalisable to other electricity markets, unlike ElecSim.

AMES [149] is an ABM specific to the US Wholesale Power Market Platform and therefore not generalizable for other countries. GAPEX [30] is an ABM framework for modelling and simulating power exchanges . GAPEX utilises an enhanced version of the reinforcement

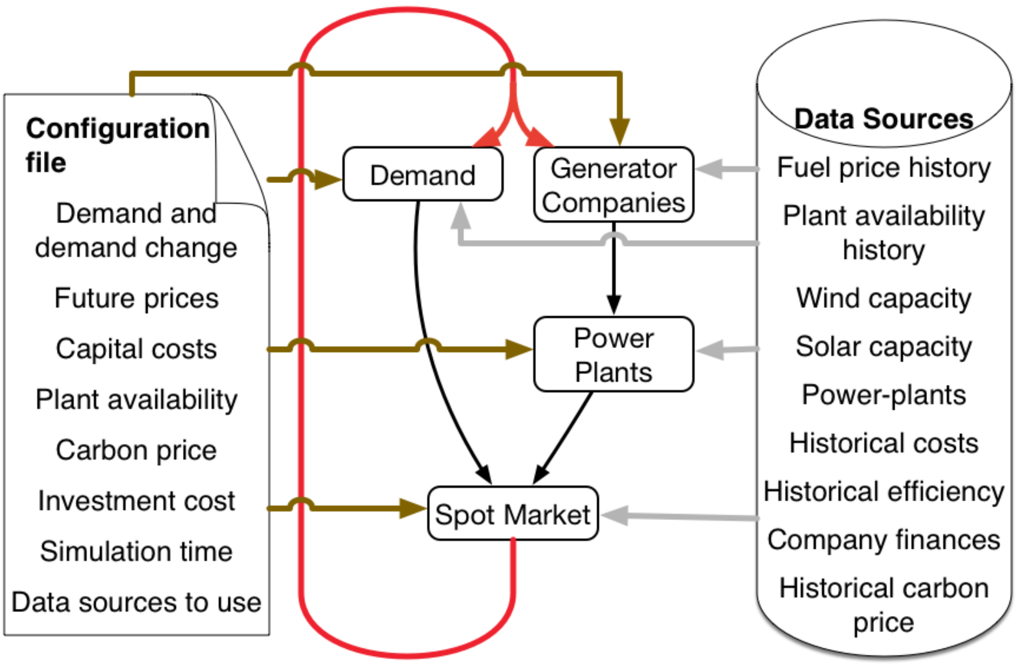


Fig. 4.1 High level overview.

technique Roth-Erev [138] to consider the presence of affine total cost functions. However, neither of these model the long-term dynamics for which ElecSim is designed.

PowerACE [139] is a closed source ABM of electricity markets that integrates short-term daily electricity trading and long-term investment decisions. PowerACE models the spot market, forward market and a carbon market. Similarly to ElecSim, PowerACE initialises GenCos with each of their power plants. However, as can be seen in Table 4.1, unlike ElecSim, PowerACE does not take into account stochasticity of price risks in electricity markets [121].

EMLab [26] is an open-source ABM toolkit for the electricity market. Like PowerACE, EMLab models an endogenous carbon market, however, they both differ from ElecSim by not taking into account stochasticity in the electricity markets, such as in outages, fuel prices and operating costs. After correspondence with the authors, however, we were unable to run the current version.

MACSEM [134] has been used to probe the effects of market rules and conditions by testing different bidding strategies. MACSEM does not model long term investments or stochastic inputs.

As can be seen from Table 4.1, none of the tools fill each of the characteristics we have defined. We therefore propose ElecSim to contribute an open source, long-term, stochastic investment model.

4.2.2 ElecSim Architecture

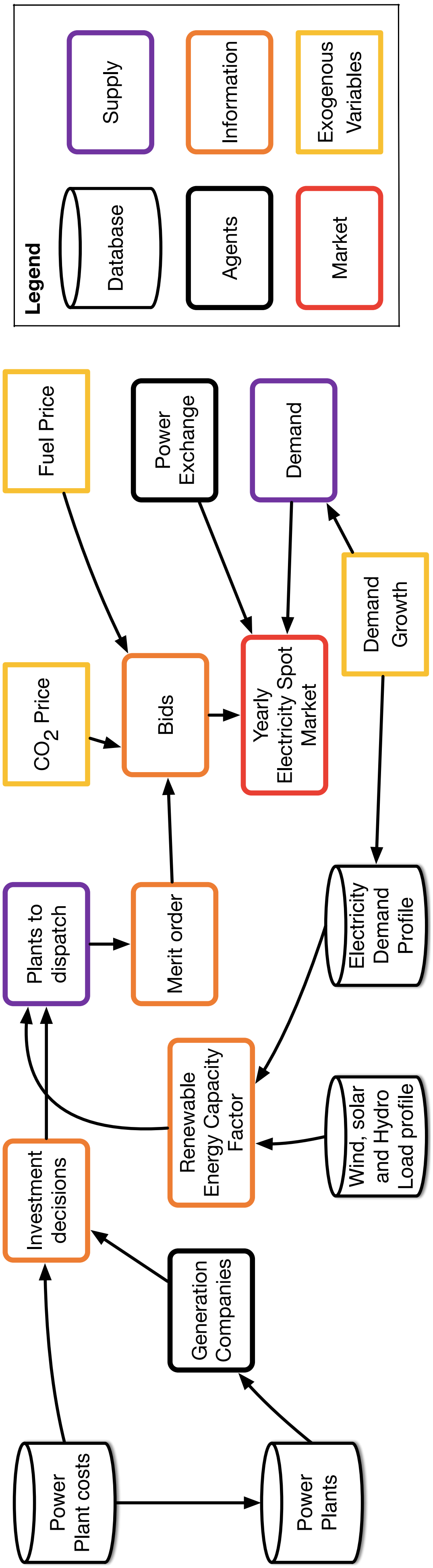


Fig. 4.2 ElecSim simulation overview

ElecSim is made up of five fundamental parts: the agents, which are split up into demand and GenCos; power plants; a Power Exchange, which controls an electricity spot market; and the data for parametrisation. A schematic of ElecSim is displayed in Figure 4.6.

Data parametrisation. ElecSim contains a configuration file and a collection of data sources for parametrisation. These data sources contain information such as historical fuel prices, historical plant availability, wind and solar capacity.

The configuration file allows for rapid changes to test different hypothesis and scenarios, and points to the different data sources. The configuration file enables one to change the demand growth and shape, future fuel and carbon prices, capital costs, plant availability, investment costs and simulation time.

Demand Agent. The demand agent is a simplified representation of aggregated demand in a country. The demand is represented as a load duration curve (LDC). An LDC is an arrangement of all load levels in descending order of magnitude. Each year, the demand agent changes each of the LDC segments proportionally.

As per Chappin *et al.* [26], we modelled the LDC of electricity demand with twenty segments. Twenty segments enabled us to capture the variation in demand throughout the year to a high degree of accuracy, whilst reducing computational complexity.

Generation Company Agents. The GenCos have two main functions. Investing in power plants and making bids to sell their generation capacity. We will first focus on the buying and selling of electricity, and then cover the investment algorithm.

The power exchange runs every year, accepting the lowest bids until supply meets demand. Once this condition is met, the spot price or system marginal price (SMP) is paid to all generators regardless of their initial bid. Generators are motivated to bid their SRMC, to ensure that their generator is being utilised, and reduce the risk of overbidding.

Investment. Investment in power plants is made based upon a net present value (NPV) calculation. NPV is a summation of the present value of a series of present and future cash flow. NPV provides a method for evaluating and comparing investments with cash flows spread over many years, making it suited for evaluating power plants which have a long lifetime.

Equation 6.1 is the calculation of NPV, where t is the year of the cash flow, i is the discount rate, N is total number of periods, or lifetime of power plant, and R_t is the net cash flow at time t.

$$NPV(i,N) = \sum_{t=0}^{N} \frac{R_t}{(1+t)^t} \tag{4.1}$$

A discount rate set by a GenCo's weighted average cost of capital (WACC) is often used [106]. WACC is the rate that a company is expected to pay on average for its stock and debt. Therefore to achieve a positive NPV, an income larger than the WACC is required. However, a higher WACC is often selected to adjust for varying risk profiles, opportunity costs and rates of return. To account for these differences we sample from a Gaussian distribution, giving us sufficient variance whilst deviating from the expected price.

To calculate the NPV, future market conditions must be considered. For this, each GenCo forecasts N years into the future, which we assume is representative of the lifetime of the plant. As in the real world, GenCos have imperfect information, and therefore must forecast expected demand, fuel prices, carbon price and electricity sale price. This is achieved by fitting functions to historical data. Each GenCo is different in that they will use differing historical time periods of data for forecasting.

Fuel and carbon price are forecast using linear regression. Demand, however, is forecast using an exponential function, which considers compounded growth. Linear regression is used if an exponential function is found to be sub-optimal.

This forecasted data is then used to simulate a market N years into the future using the electricity market algorithm. We simulate a market based on the expected bids – based on SRMC – that every operating power plant will make. This includes the removal of plants that will be past their operating period, and the introduction of plants that are in construction or pre-development stages.

There may be scenarios where demand is forecast to grow significantly, and limited investments have yet been made to meet that demand. The expected price, would be that of lost load. Lost load is defined as the price customers would be willing to pay to avoid disruption in their electricity supply. To avoid GenCos from estimating large profits, and under the assumption that further power plant investments will be made, the lost load price is replaced with a predicted electricity price using linear regression based on prices at lower points of the demand curve. If zero segments of demand are met, then the lost load price is used to encourage investment.

Once this data has been forecasted, the NPV can be calculated. GenCos must typically provide a certain percentage of upfront capital, with the rest coming from investors in the form of stock and shares or debt (WACC). The percentage of upfront capital can be customised by the user in the configuration file. The GenCos then invest in the power plants with the highest NPV.

Power Plant Parameters. Costs form an important element of markets and investment, and publicly available data for power plant costs for individual countries can be scarce. Thus, extrapolation and interpolation is required to estimate costs for power plants of differing sizes, types and years of construction.

Users are able to initialise costs relevant to their particular country by providing detailed cost parameters. They can also provide an average cost per MWh produced over the lifetime of a plant, known as levelised cost of electricity (LCOE).

The parameters used to initialise the power plants are detailed in this section. Periods have units of years and costs in £/MW unless otherwise stated: Efficiency (η) is defined as the percentage of energy from fuel that is converted into electrical energy (%). Operating period (OP) is the total period in which a power plant is in operation. Pre-development period (P_D) and pre-development costs (P_C) include the time and costs for pre-licensing, technical and design, as well as costs incurred due to regulatory, licensing and public enquiry. The construction period (C_D) and construction costs (C_C) are incurred during the development of the plant, excluding network connections. The infrastructure costs (I_C) are the costs incurred by the developer in connecting the plant to the electricity or gas grid (£). Fixed operation & maintenance costs (F_C) are costs incurred in operating the plant that do not vary based on output. Variable operation & maintenance (V_C) costs are incurred in operating the plant that depend on generator output [113].

Precise data is not available for every plant size. Linear interpolation is used to estimate individual prices between known points. When the plant to be estimated falls outside of the range of known data points, the closest power plant is used. We experimented with extrapolation but this would often lead to unrealistic costs.

If specific parameters are not known the LCOE can be used for parameter estimation, through the use of linear optimisation. Constraints can be set by the user, enabling, for example, varying operation and maintenance costs per country as a fraction of LCOE.

To fully parametrise power plants, availability and capacity factors are required. Availability is the percentage of time that a power plant can produce electricity. This can be reduced by forced or planned outages. We integrate historical data to model improvements in reliability over time.

The capacity factor is the actual electrical energy produced over a given time period divided by the maximum possible electrical energy it could have produced. The capacity factor can be impacted by regulatory constraints, market forces and resource availability. For example, higher capacity factors are common for photovoltaics in the summer, and lower in winter.

To model the intermittency of wind and solar power we allow them to contribute only a certain percentage of their total capacity (nameplate capacity) for each load segment. This percentage is based upon empirical wind and solar capacity factors. In this calculation we consider the correlation between demand and renewable resources. We are unable to model short-term storage due to ElecSim taking a single time-step per year.

When initialised, V_C is selected from a uniform distribution, with the ability for the user to set maximum percentage increase or decrease. A uniform distribution was chosen to capture the large deviations that can occur in V_C, especially over a long time period.

Fuel price is controlled by the user, however, there is inherent volatility in fuel price. To take into account this variability, an ARIMA [162] model was fit to historical gas and coal price data. The standard deviation of the residuals was used to model the variance in price that a GenCo will buy fuel in a given year. This considers differences in chance and hedging strategies.

Figure 4.2 demonstrates the simulation and how it co-ordinates runs. The world contains data and brings together GenCos, the Power Exchange and demand. The investment decisions are based on future demand and costs, which in turn influence bids made.

Exogenous variables include fuel and CO_2 prices as well as demand growth. Once the data is initialised, the world calls on the Power Exchange to operate the yearly electricity spot market. The world also settles the accounts of the GenCos, by paying bids, and removing operating and capital costs as well as loans and dividends.

Validation Validation of models is important to ascertain that the output is accurate. However, it should be noted that these long-term simulations are not predictions of the future, rather possible outcomes based upon certain assumptions. Jager posits that a certain outcome or development path, captured by empirical data, might have developed in a completely different direction due to chance. However, the processes that emerge from a model should be realistic and in keeping with expected behaviour [95].

We begin by comparing the price duration curve in the year 2018. Figure 4.3 shows the N2EX Day Ahead Auction Prices of the UK [66], the Monte-Carlo simulated electricity prices, and the non Monte-Carlo electricity price throughout the year 2018. Fuel prices varying throughout a

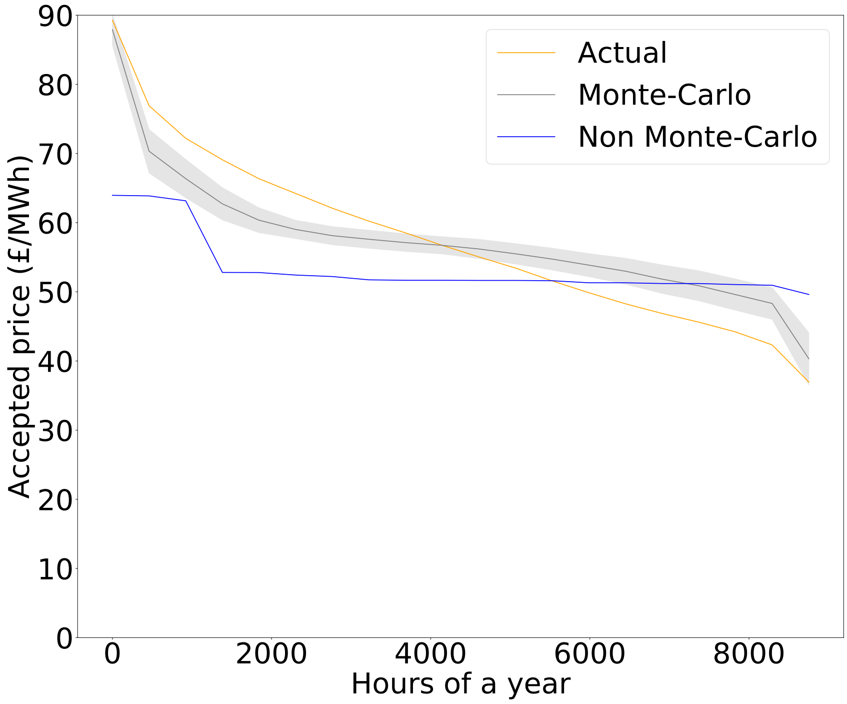


Fig. 4.3 Price duration curve which compares real electricity prices to those paid in ElecSim (2018).

Metric	N2EX Day Ahead	ElecSim	Non Monte-Carlo
Avg. Price (£/MWh)	57.49	57.52	53.39
Std. dev (£/MWh)	-	9.64	-
MAE (£/MWh)	-	3.97	8.35
RMSE (£/MWh)	-	4.41	10.2

Table 4.2 Validation performance metrics.

year, as does V_C and WACC. WACC is sampled from a Gaussian distribution with a standard deviation of $\pm 3\%$. V_C is sampled from a uniform distribution between 30% and 200% of the mean V_C price, whilst fuel price is sampled from the residuals of an ARIMA model fit on historical data. The N2EX Day Ahead Market is a day ahead market run by Nord Pool AS. Nord Pool AS runs the largest market for electrical energy in Europe, measured in volume traded and in market share [66].

We ran the initialisation of the model 40 times to capture the price variance. Outliers were removed as on a small number of occasions large jumps in prices at peak demand occurred which deviated from the mean. We did this, as although this does occur in real life, it occurs at a smaller fraction of the time than 5% of the year (modelled LDC), therefore the results would be unreasonably skewed for the highest demand segment.

Figure 4.3 demonstrates very little variance in the non-stochastic case. This is due to the fact that combined cycle gas turbines (CCGTs) set the spot price. These CCGTs have little variance between one another as they were calibrated using the same dataset. By adding stochasticity of fuel prices and operation and maintenance prices, a curve that more closely resembles the actual data occurs. The stochastic curve, however, does not perfectly fit the real data, which may

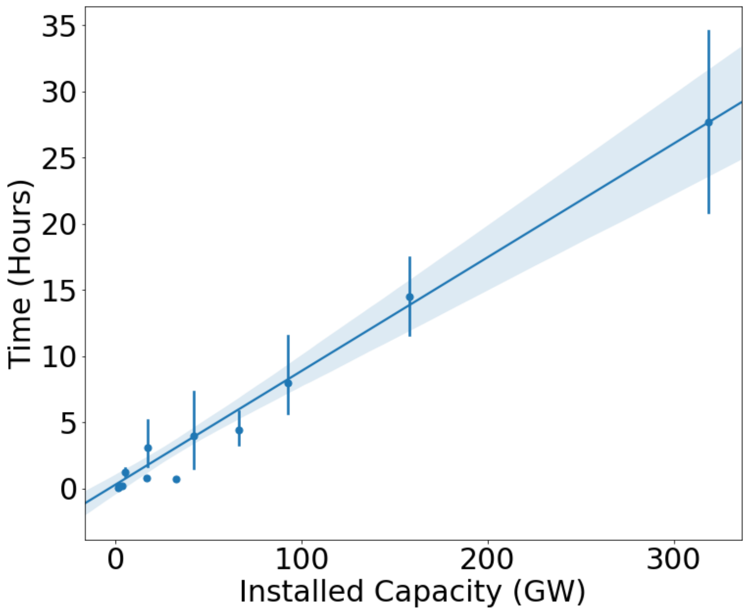


Fig. 4.4 Run times of different sized countries.

be due to higher variance in fuel prices and historical differences in operation and maintenance costs between power plants. One method of improving this would be fitting the data used to parametrise to the curve.

Table 4.2 shows performance metrics of the stochastic and non-stochastic runs versus the actual price duration curve . The stochastic implementation, improves the mean absolute error (MAE) of the non-stochastic case by 52.5%.

By observing the processes that emerge from the long-term scenarios, we can see that carbon price and investment in renewable generation are positively correlated, as would be expected. The highest NPV calculations were for onshore wind and CCGT plants. This is realistic for the United Kingdom, where subsidies are required for other forms of generation such as coal and nuclear.

Performance Figure 4.4 shows the running time for ElecSim with varying installed capacity. We varied demand between 2GW and 320GW to see the effect of different sized countries on running time. The makeup of the electricity mix was achieve through stratified sampling of the UK electricity mix. The results show a linear time complexity.

4.3 Scenario Testing

Here we present example scenario runs using ElecSim. We vary the carbon tax and grow or reduce total electricity demand. This enables us to observe the effects of carbon tax on investment. In this paper we have presented scenarios where electricity demand decreases 1% per year, due to the recent trend in the UK.

For the first scenario run displayed, we have approximated the predictions by the UK Government, where carbon tax increases linearly from £18 to £200 by 2050 [41]. Figure 4.5a demonstrates a significant increase in gas turbines in the first few years, followed by a decrease, with onshore wind increasing.

Figure 4.5b displays a run with a £40 carbon tax. This run demonstrates a higher share of onshore wind than in the previous scenario.

These runs demonstrate that a consistent, but relatively low carbon tax can have a larger impact in the uptake of renewable energy than increasing carbon tax over a long time frame. We hypothesise that an early carbon tax affects the long-term dynamics of the market for many years. We, therefore, suggest early action on carbon tax to transition to a low-carbon energy supply

4.3.1 Conclusion

Liberalised electricity markets with many heterogenous players are suited to be modelled with ABMs. ABMs incorporate imperfect information as well as heterogeneous actors. ElecSim models imperfect information through forecasting of electricity demand and future fuel and electricity prices. This leads to agents taking risk on their investments, and model market conditions more realistically.

We demonstrated that increasing carbon tax can lead to an increase in investment of low-carbon technologies. We showed that early decisions have a long-term impact on the energy mix.

Our future work includes comparing agent-learning techniques, using multi-agent reinforcement learning algorithms to allow agents to learn in a non-static environment. We propose the integration of a higher temporal and spatial resolution to model changes in daily demand, as well as capacity factors by region, and transmission effects. This will allow us to model that demand is met at all times and not just on average.

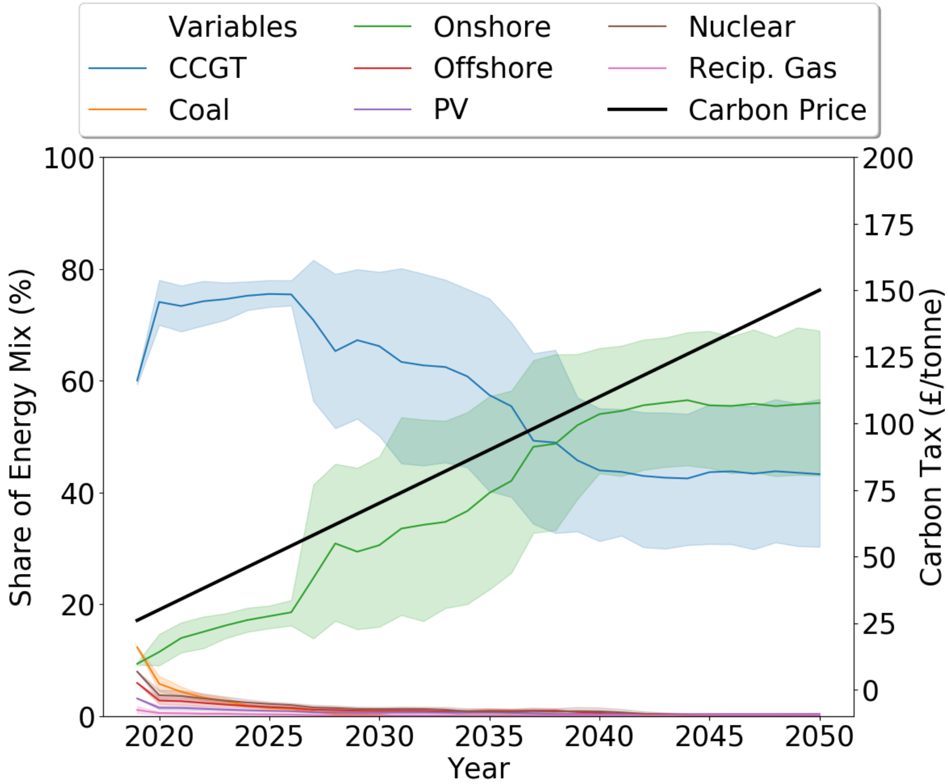
4.4 e-Energy 2019 Poster

4.4.1 Introduction

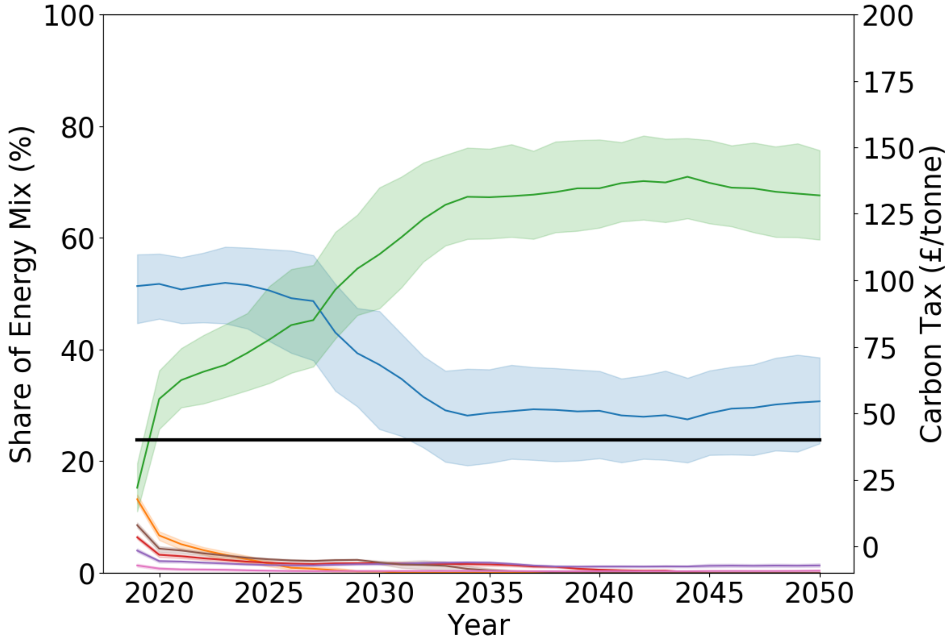
Governmental policy is a tool that can be used to aid in the transition to a low-carbon economy to prevent the worst effects of climate change. Options include a tax on all carbon emissions or subsidies in low-carbon technologies. In this paper, we vary carbon taxes to assess the long-term impacts on investment in the electricity market. We used a general agent-based model simulation made for wholesale electricity markets, created by us named ElecSim.

Simulation is a technique to create a physical system in a virtual world. In this context a model is defined as a set of mathematical formulas and algorithms which are designed to mimic real life [54]. Simulation allows practitioners to rapidly prototype high risk ideas in this virtual model and assess their outcome before implementation in the real world.

The electricity market in many western democracies consists of multiple heterogenous actors acting for their own best interest [121]. Agent-based modelling is a technique which allows for the simulation of these heterogenous actors with different risk profiles, profit requirements and preferences. A number of agent-based models have been used to model the impact of carbon



(a) £26 to £150 linearly increasing carbon tax.



(b) £40 carbon tax

Fig. 4.5 Scenarios with varying carbon taxes and decreasing demand (-1%/year)

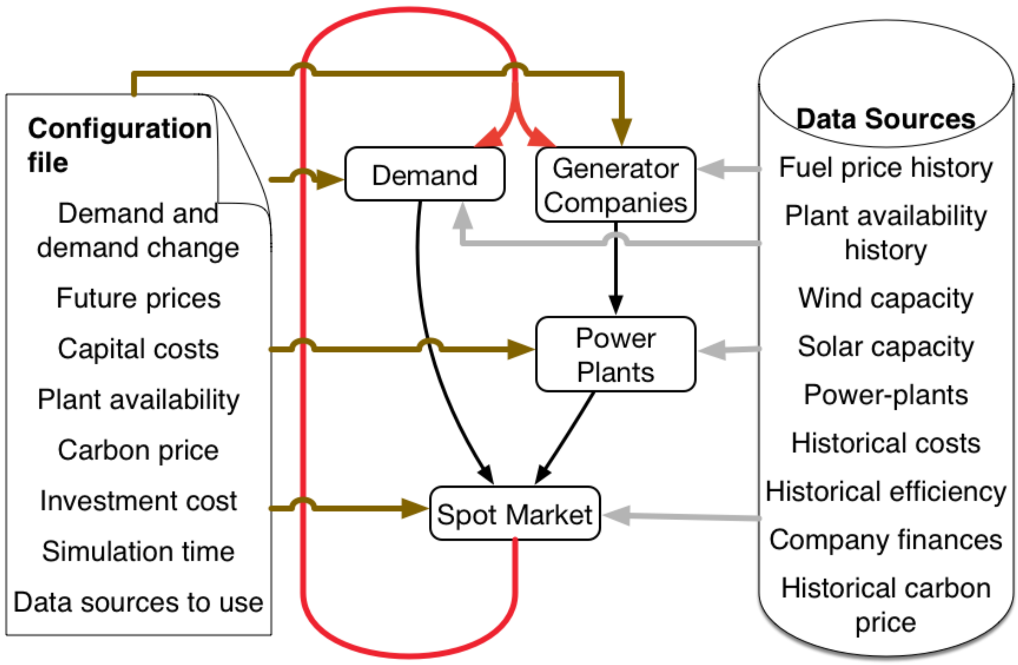


Fig. 4.6 System overview of agent-based market model.

tax on long term investments [151, 29, 26]. ABMs have been utilised in this field to address phenomena such as market power [137].

We model the realisation of the wholesale electricity market in the United Kingdom and adjust carbon tax in our agent-based model to see the effect of long-term investment. Whilst we have modelled the United Kingdom, it would be possible to model for any country with different parameters. We posit that decisions made today can have complex long-term consequences, the process of which can be observed through simulation.

This paper details our model and different carbon scenarios. Section **??** details the model, assumptions made and parameters. Section **??** presents our results. We conclude our work in Section **??** and explore possible routes forward .

4.4.2 Model architecture

The agent-based model is made up of five significant parts: the agents which are made up of the generation companies (GenCos) and demand agents; power plants and a market operator which controls the spot market. How these parts interact are displayed in Figure 4.6. The relevant data sources are also provided there.

We initialise the United Kingdom with our model with exemplar data from the UK. We model every single power plant in operation in the year 2018, which are owned by their respective generation companies. Individual historical power plant costs are estimated from levelized cost of electricity (LCOE) [38, 90, 93], whereas future and present power plant costs are taken from the department of business and industrial strategy [41]. The variable operation and maintenance cost was defined stochastically to model the varying costs per project. A uniform distribution was chosen to provide sufficient variance between projects.

The demand agent is modelled as a single aggregated demand, split up into 20 segments of a yearly load duration curve (LDC), enabling us to increase speed of computation whilst maintaining accuracy. An LDC is defined as load sorted in order of magnitude.

We model the influence of outages using availability data for gas, coal, photovoltaic, and wind power generators [113, 88, 23]. Historical availabilities are modelled for old gas, coal and hydro power plants [9]. Capacity factors per geographical location were taken as an average of the UK for solar and wind [131, 148]. Where a capacity factor is defined as the ratio of electrical output over a given time period over the maximum possible electrical energy output.

The generation companies make electricity bids each year for each of their power plants. The market operator then matches demand with supply in order of price, also known as merit-order dispatch. We model a uniform pricing market, where each of the companies are paid the highest accepted bid per load segment.

GenCos have the ability to invest every year in new power plants based on the expected net present value (NPV) of each type of power plant. NPV is a summation of the present value of a series of present and future cash flow. The NPV calculation is dependent on a stochastic representation of GenCos predictions of fuel, carbon and electricity price and demand.

Each GenCo has a separate weighted average cost of capital (WACC), which is the average rate that a company is expected to for its stock and debt. This is used as the discount rate in the NPV calculation [106]. The WACC is modelled as a stochastic variable, with a Gaussian distribution, with a $\pm 3\%$ standard deviation, with values of 5.9% for non-nuclear power plants, and 10% for nuclear power plants [108, 129].

The model took yearly time-steps to limit the impact on computation time, however, to model the intermittency of renewable generation, we correlated demand with the respective capacity factor, enabling for example, solar and wind to only contribute a certain capacity to their load curve.

Stochasticity of fuel price within a year was also modelled, to take into account difference in hedging strategies and chance. An ARIMA model [162] was fit to historic coal and natural gas prices.

4.4.3 Results

We experimented with the following levels of carbon tax: £10 ($13), £20 ($26) and £70 ($90) with demand decreasing 1% per year. This was chosen due to the increasing efficiency of homes, industry and technology, and due to the recent trend in the UK. We run each scenario 8 times to capture the stochastic nature of the process. Via the observation of the emergent investment behaviour until 2050, an understanding of how real life investors may behave emerges.

Figure 4.7a shows that with a carbon tax of £10, whilst renewable technology does grow, gas power plants provide the majority of supply in each year. However, at a level of £20 the increase in wind turbines is enough to match gas turbines. A carbon tax of £70, however, shows a near 100% uptake of wind turbines.

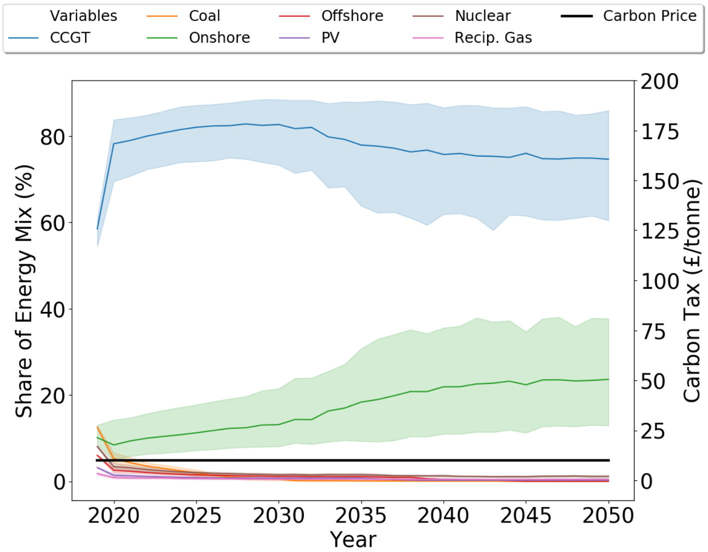
It is infeasible for the power supply to be provided solely by wind turbines today. This overestimation, however, is due to the low time granularity of the model [32]. This scenario therefore assumes perfect storage capabilities.

4.4.4 Conclusion

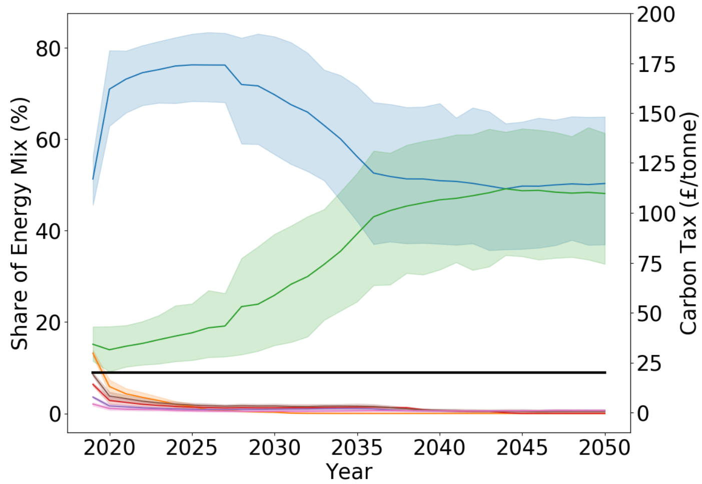
Agent-based models provide a method of simulating investor behaviour in an electricity market. We observed that an increase in carbon tax had a significant impact on investment. These findings enable policy makers to better understand the impact that their decisions may have. For a high uptake of renewable energy technology, rapid results can be seen after 10 years with a carbon tax of £70 ($90).

Agent-based models open up the possibility of testing differing investor behaviours through techniques such as reinforcement learning. This can be extended to incorporate collusion which can have an impact in liberalized electricity markets [16].

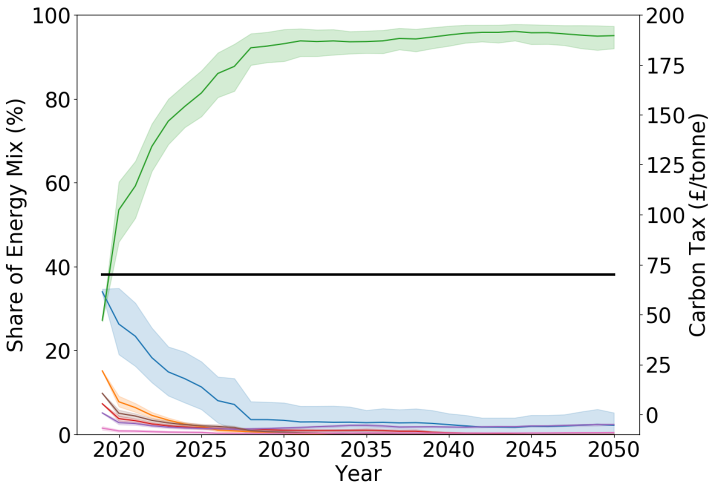
4.5 e-Energy 2020 Paper



(a) £10 carbon tax.



(b) £20 carbon tax.



(c) £70 carbon tax.

Fig. 4.7 Scenarios from 2020 to 2050 with varying carbon tax.

Abstract

Electricity market modelling is often used by governments, industry and agencies to explore the development of scenarios over differing timeframes. For example, what would the reduction in cost of renewable energy mean for investments in gas power plants or what would be an optimum strategy for carbon tax or subsidies?

Optimization based solutions are the dominant approach for analysing energy policy. However, these types of models have certain limitations such as the need to be interpreted in a normative manner, and the assumption that the electricity market remains in equilibrium throughout. Through this work, we show that agent-based models are a viable technique to simulate decentralised electricity markets. The aim of this paper is to validate an agent-based modelling framework to increase confidence in its ability to be used in policy and decision making.

Our framework is able to model heterogenous agents with imperfect information. The model uses a rules-based approach to approximate the underlying dynamics of a real life, decentralised electricity market. We use the UK as a case-study, however our framework is generalisable to other countries. We increase the temporal granularity of the model by selecting representative days of electricity demand and weather using a *k*-means clustering approach.

We show that our modified framework, ElecSim, is able to adequately model the transition from coal to gas observed in the UK between 2013 and 2018. We are also able to simulate a future scenario to 2035 which is similar to the UK Government, Department for Business and Industrial Strategy (BEIS) predictions, showing a more realistic increase in nuclear power over this time period. This is due to the fact that with current, large nuclear technology, electricity is generated almost instantaneously and has a low relative short-run marginal cost [41]. This low short-run marginal cost means that new nuclear will be dispatched on the electricity market once it has been turned on, and thus will not increase gradually.

4.6 Introduction

Impacts on natural and human systems due to global warming have already been observed, with many land and ocean ecosystems having changed. A rise in carbon emissions increases the risk of severe impacts on the world such as rising sea levels, heat waves and tropical cyclones [120]. A study by Cook *et al.* demonstrated that 97% of scientific literature concurred that recent global warming was anthropogenic [34]. Limiting global warming requires limiting the total cumulative global anthropogenic emissions of CO_2 [120].

Global carbon emissions from fossil fuels, however, have significantly increased since 1900 [17]. Fossil-fuel based electricity generation sources such as coal and natural gas currently provide 65% of global electricity. Low-carbon sources such as solar, wind, hydro and nuclear provide 35% [20]. To halt this increase in CO_2 emissions, a transition of the energy system towards a renewable energy system is required.

However, such a transition needs to be performed in a gradual and non-disruptive manner. This ensures that there are no electricity shortages or power cuts that would cause damage to businesses, consumers and the economy.

To ensure such a transition, energy modelling is often used by governments, industry and agencies to explore possible scenarios under different variants of government policy, future electricity generation costs and energy demand. These energy modelling tools aim to mimic the behaviour of energy systems through different sets of equations and data sets to determine the energy interactions between different actors and the economy [117].

Optimization based solutions are the dominant approach for analysing energy policy [26]. However, the results of these models should be interpreted in a normative manner. For example, how investment and policy choices should be carried out, under certain assumptions and scenarios. The processes which emerge from an equilibrium model remain a black-box, making it difficult to fully understand the underlying dynamics of the model [26].

In addition to this, optimization models do not allow for endogenous behaviour to emerge from typical market movements, such as investment cycles [26, 65]. By modelling these naturally occurring behaviours, policy can be designed that is robust against movements away from the optimum/equilibrium. Thus, helping policy to become more effective in the real world.

The work presented in this paper builds on the ABM, ElecSim, developed by Kell *et al.* [101]. Agent-based models differ from optimization models by the fact that they are able to explore '*what-if*' questions regarding how a sector could develop under different prospective policies, as opposed to determining optimal trajectories. ABMs are particularly pertinent in decentralised electricity markets, where a centralised actor does not dictate investments made within the electricity sector. ABMs have the ability to closely mimic the real world by, for example, modelling irrational agents, in this case Generation Companies (GenCos) with incomplete information in uncertain situations [61].

There is a desire to validate the ability of energy-models to make long-term predictions. Validation increases confidence in the outputs of a model and leads to an increase in trust amongst the public and policy makers. Energy models, however, are frequently criticised for being insufficiently validated, with the performance of models rarely checked against historical outcomes [15].

In answer to this we postulate that ABMs can provide accurate information to decision makers in the context of electricity markets. We increase the temporal granularity of the work by Kell *et al.* [101] and use genetic algorithms to tune the model to observed data enabling us to perform validation. This enables us to understand the parameters required to observe certain phenomena, as well as use these fitted parameters to make inferences about the future.

We use a genetic algorithm approach to find an optimal set of price curves predicted by generation companies (GenCos) that adequately model observed investment behaviour in the real-life electricity market in the United Kingdom. Similar techniques can be employed for other countries of various sizes [101].

Similarly to Nahmmacher *et al.* we demonstrate how clustering of multiple relevant time series such as electricity demand, solar irradiance and wind speed can reduce computational time by selecting representative days [123]. In this context, representative days are a subset of days that have been chosen due to their ability to approximate the weather and electricity demand in an entire year. Distinct to Nahmacher *et al.* we use a *k*-means clustering approach [53] as opposed to a hierarchical clustering algorithm described by Ward [98]. We chose the *k*-means clustering approach due to previous success of this technique in clustering time series [102].

We measure the accuracy of projections for our improved ABM with those of the UK Government's Department for Business, Energy and Industrial Strategy (BEIS) for the UK electricity market between 2013 and 2018. In addition to this, we compare our projections from 2018 to 2035 to those made by BEIS in 2018 [42].

We are able to adequately model the transitional dynamics of the electricity mix in the United Kingdom between 2013 and 2018. During this time there was an $\sim$88% drop in coal use, $\sim$44% increase in Combined Cycle Gas Turbines (CCGT), $\sim$111% increase in wind energy and increase in solar from near zero to $\sim$1250MW. We are therefore able to test our model in a transition of sufficient magnitude.

We show in this paper that agent-based models are able to adequately mimic the behaviour of the UK electricity market under the same specific scenario conditions. Concretely, we show that under an observed carbon tax strategy, fuel price and electricity demand, the model ElecSim closely matches the observed electricity mix between 2013 and 2018. We achieve this by determining an exogenous predicted price duration curve using a genetic algorithm to minimise error between observed and simulated electricity mix in 2018. The predicted price curve is an arrangement of all price levels in descending order of magnitude. The predicted price duration curve achieved is similar to that of the simulated price duration curve in 2018, increasing confidence in the underlying dynamics of our model.

In addition, we compare our projections to those of the BEIS reference scenario from 2018 to 2035 [42]. To achieve this we use the same genetic algorithm optimisation technique as during our validation stage, optimising for predicted price duration curves. Our model demonstrates that we are able to closely match the projections of BEIS by finding a set of realistic price duration curves which are subject to investment cycles. Our model, however, exhibits a more realistic step change in nuclear output than that of BEIS. This is because, whilst BEIS projects a gradual increase in nuclear output, our model projects that nuclear output will grow instantaneously at a single point in time as a new nuclear power plant comes online.

This allows us to verify the scenarios of other models, in this case BEIS' reference scenario, by ascertaining whether the optimal parameters required to achieve such scenarios are realistic. In addition to this, we are able to use these parameters to analyse '*what-if*' questions with further accuracy.

We increased the temporal granularity of the model using a *k*-means clustering approach to select a subset of representative days for wind speed, solar irradiance and electricity demand. This subset of representative days enabled us to approximate an entire year, and only required a fraction of the total time-steps that would be necessary to model each day of a year independently. This enabled us to decrease execution time. We show that we are able to provide an accurate framework, through this addition, to allow policy makers, decision makers and the public explore the effects of policy on investment in electricity generators.

We demonstrate that with a genetic algorithm approach we are able to optimise parameters to improve the accuracy of our model. Namely, we optimise the predicted electricity price, the uncertainty of this electricity price and nuclear subsidy. We use validation to verify our model using the observed electricity mix between 2013-2018.

The main contribution of this work is to demonstrate that it is possible for agent-based models to accurately model transitions in the UK electricity market. This was achieved by comparing our simulated electricity mix to the observed electricity mix between 2013 and 2018. In this time a transition from coal to natural gas was observed. We demonstrate that a high temporal granularity is required to accurately model fluctuations in wind and solar irradiance for intermittent renewable energy sources.

In Section 6.3 we introduce a review of techniques used for validating electricity market models as well as fundamental challenges of electricity model validation. Section 6.4 explores our approach to validate our model. In Section 6.5 we discuss the modifications made to our model to improve the results of validation. In Section 6.6 we present our results, and we conclude in Section 6.7.

4.7 Literature Review

In this section we cover the difficulties inherent in validating energy models and the approaches taken in the literature.

4.7.1 Limits of Validating Energy Models

Beckman *et al.* state that questions frequently arise as to how much faith one can put in energy model results. This is due to the fact that the performance of these models as a whole are rarely checked against historical outcomes [15].

Under the definition by Hodges *et al.* [80] long-range energy forecasts are not validatable [36]. Under this definition, validatable models must be observable, exhibit constancy of structure in time, exhibit constancy across variations in conditions not specified in the model and it must be possible to collect ample data [80].

Whilst it is possible to collect data for energy models, the data covering important characteristics of energy markets are not always measured. Furthermore, the behaviour of the human population and innovation are neither constant nor entirely predictable. This leads to the fact that static models cannot keep pace with global long-term evolution. Assumptions made by the modeller may be challenged in the form of unpredictable events, such as the oil shock of 1973 [36].

This, however, does not mean that energy-modelling is not useful for providing advice in the present. A model may fail at predicting the long-term future because it has forecast an undesirable event, which led to a pre-emptive change in human behaviour. Thus avoiding the original scenario that was predicted. This could, therefore, be viewed as a success of the model.

Schurr *et al.* argued against predicting too far ahead in energy modelling due to the uncertainties involved [156]. However, they specify that long-term energy forecasting is useful to provide basic information on energy consumption and availability which is helpful in public debate and in guiding policy makers.

Ascher concurs with this view and states that the most significant factor in model accuracy is the time horizon of the forecast; the more distant the forecast target the less accurate the model. This can be due to unforeseen changes in society as a whole [62].

It is for these reasons that we focus on a shorter-term (5-year) horizon window when validating our model. This enables us to have an increased confidence that the dynamics of the model work without external shocks and can provide descriptive advice to stakeholders. However, it must be noted that the UK electricity market exhibited a fundamental transition from natural gas to coal electricity generation during this period, meaning that a simple data-driven modelling approach would not work.

In addition to this short-term cross-validation, we compare our long-term projections to those of BEIS from 2018 to 2035. It is possible that our projections and those of BEIS could be wrong, however, this allows us to thoroughly test a particular scenario with different modelling approaches, and allow for the possibility to identify potential flaws in the models.

4.7.2 Validation Examples

In this section we explore a variety of approaches used in the literature for energy model validation.

The model OSeMOSYS [85] is validated against the similar model MARKAL/TIMES through the use of a case study named UTOPIA. UTOPIA is a simple test energy system bundled with ANSWER, a graphical user interface packaged with the MARKAL model generator [89, 126]. Hunter *et al.* use the same case study to validate their model Temoa [89]. In these cases, MARKAL/TIMES is seen as the "gold standard". In this paper, however, we argue that the ultimate gold standard should be real-world observations, as opposed to a hypothetical scenario.

The model PowerACE demonstrates that realistic prices are achieved by their modelling approach, however, they do not indicate success in modelling GenCo investment over a prolonged time period [136].

Barazza *et al.* validate their model, BRAIN-Energy, by comparing their results with a few years of historical data, however, they do not compare the simulated and observed electricity mix [13].

Work by Koomey *et al.* expresses the importance of conducting retrospective studies to help improve models [107]. In this case, a model can be rerun using historical data in order to determine how much of the error in the original forecast resulted from structural problems in the model itself, or how much of the error was due to incorrect specification of the fundamental drivers of the forecast [107].

A retrospective study published in 2002 by Craig *et al.* focused on the ability of forecasters to accurately predict electricity demand from the 1970s [36]. They found that actual energy usage in 2000 was at the very lowest end of the forecasts, with only a single exception. They found that these forecasts underestimated unmodelled shocks such as the oil crises which led to an increase in energy efficiency.

Hoffman *et al.* also developed a retrospective validation of a predecessor of the current MARKAL/TIMES model, named Reference Energy System [82], and the Brookhaven Energy System Optimization Model [2]. These were studies applied in the 70s and 80s to develop projections to the year 2000. This study found that the models had the ability to be descriptive, but were not entirely accurate in terms of predictive ability. They found that emergent behaviours in response to policy had a strong impact on forecasting accuracy. The study concluded that forecasts must be expressed in highly conditioned terms [81].

4.8 Problem Formulation

In this section we detail the approach taken in this paper to validate our model, including the parameters used for optimization.

Specifically, we use a genetic algorithm to find the predicted price duration curves which lead to the smallest error between our simulated electricity mix and the scenarios tested. The scenarios examined here are the observed electricity mix of the UK between 2013 and 2018 and the BEIS reference scenario projected in 2018 till 2035. When projecting the BEIS reference scenario we also optimise for nuclear subsidy and uncertainty in the price duration curves.

4.8.1 Optimization Variables

For GenCos to adequately make investments, they must formulate an expectation of future electricity prices over the lifetime of a plant. For this paper, we use the net present value (NPV) metric to compare investments.

NPV provides a method for evaluating and comparing investments with cash flows spread over many years, making it suited for evaluating power plants which have a long lifetime.

Equation 6.1 is the calculation of NPV, where t is the year of the cash flow, i is the discount rate, N is the total number of years, or lifetime of power plant, and R_t is the net cash flow of the year t.

The net cash flow, R_t, is calculated by subtracting both the operational and capital costs from revenue over the expected lifetime of the prospective plant. The revenue gained by each prospective plant is the expected price they will gain per expected quantity of MWh sold over the expected lifetime of the plant. This is shown formally in Equation 6.2:

$$NPV(i,N) = \sum_{t=0}^{N} \frac{R_t}{(1+t)^t} \tag{4.2}$$

$$R_t = \sum_{t=0}^{T} \sum_{h=0}^{H} \sum_{m=0}^{M} \big(m_{h,t}(PPDC_{h,t} - C_{var_{h,t}})\big) - C_c \tag{4.3}$$

where $m_{h,t}$ is the expected quantity of megawatts sold in hour h of year t. $PPDC_{h,t}$ is the predicted price duration curve at year t and hour h. $C_{var_{h,t}}$ is the variable cost of the power plant, which is dependent on expected megawatts of electricity produced, C_c is the capital cost.

The predicted price duration curve ($PPDC_{h,t}$) is an expectation of future electricity prices over the lifetime of the plant. The $PPDC_{h,t}$ is a function of supply and demand. However, with renewable electricity generator costs falling, future prices are uncertain and largely dependent upon long-term scenarios of electricity generator costs, fuel prices, carbon taxes and investment decisions. [92]. Due to the uncertainty of future electricity prices over the horizon of the lifetime of a power plant we have set future electricity prices as an exogenous variable that can be set by the user in ElecSim.

To gain an understanding of expected electricity prices that lead to particular scenarios we use a genetic algorithm optimisation approach. This enables us to understand the range of future electricity prices that lead to certain scenarios developing. In addition, it allows us to understand whether the parameters required for certain scenarios to develop are realistic. This enables us to check the assumptions of our model and the likelihood of scenarios.

Further, using these optimised parameters, we are better able to further explore '*what-if*' scenarios.

4.8.2 Validation with Observed Data

To verify the accuracy of the underlying dynamics of ElecSim, the model was initialised to data available in 2013 and allowed to develop until 2018. We used a genetic algorithm to find the optimum price duration curve predicted (*PPDC*) by the GenCos 10 years ahead of the year of the simulation. This *PPDC* was used to model expected rate of return of prospective generation types, as shown in Equations 6.1 and 6.2.

The genetic algorithm's objective was to reduce the error of simulated and observed electricity mix in the year 2018 by finding a suitable *PPDC* used by each of the GenCos for investment evaluation.

Scenario

For this experiment, we initialised ElecSim with parameters known in 2013 for the UK. ElecSim was initialised with every power plant and respective GenCo that was in operation in 2013 using the BEIS DUKES datatset [51]. The funds available to each of the GenCos was taken from publicly released official company accounts at the end of 2012 [40].

To ensure that the development of the electricity market from 2013 to 2018 was representative of the actual scenario between these years, we set the exogenous variables, such as carbon and fuel prices, to those that were observed during this time period. In other words, the scenario modelled equated to the observed scenario.

The data for the observed EU Emission Trading Scheme (ETS) price between 2013 and 2018 was taken from [140]. Fuel prices for each of the fuels were taken from [39]. The electricity load data was modelled using data from [gri], offshore and onshore wind and solar irradiance data was taken from [131]. There were three known significant coal plant retirements in 2016. These were removed from the simulation at the beginning of 2016.

Optimisation problem

The price duration curve was modelled linearly in the form $y = mx + c$, where y is the cost of electricity, m is the gradient, x is the demand of the price duration curve and c is the intercept.

Equation 6.3 details the optimisation problem formally:

$$\min_{m,c} \sum_{o \in O} \left(\frac{|A_o - f_o(m,c)|}{||O||} \right) \tag{4.4}$$

where $o \in O$ refers to the average percentage electricity mix during 2018 for wind (both offshore and onshore generation), nuclear, solar, CCGT, and coal, where O refers to the set of these values. A_o refers to observed electricity mix percentage for the respective generation type in 2018. $f_o(m,c)$ refers to the simulated electricity mix percentage for the respective generation type, also in 2018. The input parameters to the simulation are m and c from the linear *PPDC*, previously discussed, ie. $y = mx + c$. $||O||$ refers to the cardinality of the set.

4.8.3 Long-Term Scenario Analysis

In addition to verifying the ability for ElecSim to mimic observed investment behaviour over 5 years, we compared ElecSim's long-term behaviour to that of the UK Government's Department for Business, Energy and Industrial Strategy (BEIS) [42]. This scenario shows the projections of generation by technology for all power producers from 2018 to 2035 for the BEIS reference scenario. This is the same scenario as discussed in the next section, 6.4.3.

Scenario

We initialised the model to 2018 based on [101]. The scenario for development of fuel prices and carbon prices were matched to that of the BEIS reference scenario [42].

Optimisation problem

The optimisation approach taken was a similar process to that discussed in Sub-Section 6.4.2, namely using a genetic algorithm to find the optimum expected price duration curve. However, instead of using a single expected price duration curve for each of the agents for the entire simulation, we used a different expected price duration curve for each year, leading to 17 different curves. This enabled us to model the non-static dynamics of the electricity market over this extended time period.

In addition to optimising for multiple expected price duration curves, we optimised for a nuclear subsidy, S_n. Further, we optimised for the uncertainty in the expected price parameters m and c, named σ_m and σ_c respectively, where σ is the standard deviation in a normal distribution. m and c are the parameters for the predicted price duration curve, as previously defined, of the form $y = mx + c$.

This enabled us to model the different expectations of future price curves between the independent GenCos. The addition of a nuclear subsidy as a parameter is due to the likely requirement for Government to provide subsidies for new nuclear [150].

A modification was made to the reward algorithm for the long-term scenario case. Rather than using the discrepancy between observed and simulated electricity mix in the final year (2018) as the reward, a summation of the error metric for each simulated year was used. This is detailed formally in Equation 6.4:

$$\min_{m \in M, m \in C} \sum_{y \in Y} \sum_{o \in O} \left(\frac{\left| A_{y_o} - f_{y_o}(m_y, c_y) \right|}{||O||} \right) \tag{4.5}$$

where M and C are the sets of the 17 parameters of m_y and c_y respectively for each year, y. $y \in Y$ refers to each year between 2018 and 2035. m_y and c_y refer to the parameters for the predicted price duration curve, of the form $y = mx + c$ for the year y. A_{y_o} refers to the actual electricity mix percentage for the year y and generation type o. Finally, $f_{y_o}(m_y, c_y)$ refers to the simulated electricity mix percentage with the input parameters to the simulation of m and c for the year y.

4.9 Implementation details

In this section we discuss the changes made to Kell *et al.* to improve the results of validation [101]. Further, we introduce the genetic algorithm used to find the optimal parameter sets.

ElecSim is made up of six distinct sections: 1) power plant data; 2) scenario data; 3) the time-steps of the algorithm; 4) the power exchange; 5) the investment algorithm and 6) the generation companies (GenCos) as agents. ElecSim has been previously published [101], however, we have made amendments to the original work in the form of efficiency improvements to decrease compute time as well as increase the granularity of time-steps from yearly to representative days. Representative days, in this context, are a subset of days which when scaled up to 365 days can adequately represent a year.

In this paper, we initialised the model to a scenario of the United Kingdom as an example, however, the fundamental dynamics of the model remain the same for other decentralised electricity markets. In this section we detail the modifications we made to ElecSim for this paper. Further details of the design decisions of ElecSim are discussed in [101].

4.9.1 Representative Days

In previously published work, ElecSim modelled a single year as 20 time-steps for solar irradiance, onshore and offshore wind and electricity demand [101]. Similarly to findings of other authors, this relatively low number of time-steps led to an overestimation of the uptake of intermittent renewable energy resources (IRES) and an underestimation of flexible technologies [115, 72]. This is due to the fact that the full intermittent nature of renewable energy could not be accurately modelled in such a small number of time-steps.

To address this problem, whilst maintaining a tractable execution time, we approximated a single year as a subset of proportionally weighted, representative days. This enabled us to reduce computation time. Each representative day consisted of 24 equally separated time-steps, which model hours in a day. Hourly data was chosen, as this was the highest resolution of the dataset available for offshore and onshore wind and solar irradiance [131]. A lower resolution would allow us to model more days, however, we would lose accuracy in terms of the variability of the renewable energy sources.

Similarly to Nahmmacher *et al.* we used a clustering technique to split similar days of weather and electricity demand into separate groups. We then selected the historic day that was closest to the centre of the cluster, known as the medoid, as well as the average of the centre, known as the centroid [123]. Distinct to Nahmmacher, however, we used the *k*-means clustering algorithm [53] as opposed to the Ward's clustering algorithm [98]. This was due to the ability for the *k*-means algorithm to cluster time-series into relevant groups [103]. These days were scaled proportionaly to the number of days within their respective cluster to approximate a total of 365 days.

Equation 6.5 shows the series for a medoid or centroid, selected by the *k*-means algorithm:

$$P_h^{x,i} = \{P_1, P_2, \ldots, P_{24}\} \tag{4.6}$$

where $P_h^{x,i}$ is the medoid for series x, where $x \in X$ refers to offshore wind capacity factor, onshore wind capacity factor, solar capacity factor and electricity demand, h is the hour of the day and i is the respective cluster. $\{P_1, P_2, \ldots, P_{24}\}$ refers to the capacity values at each hour of the representative day.

We then calculated the weight of each cluster. This gave us a method of assigning the relative importance of each representative day when scaling the representative days up to a year. The weight is calculated by the proportion of days in each cluster. This gives us a method of determining how many days within a year are similar to the selected medoid or centroid. The calculation for the weight of each cluster is shown by Equation 6.6:

$$w_i = \frac{n_i}{||N||} \tag{4.7}$$

where w_i is the weight of cluster i, n_i is the number of days in cluster i, and $||N||$ is the total number of days that have been used for clustering.

The next step was to scale up the representative days to represent the duration curve of a full year. We achieved this by using the weight of each cluster, w_i, to increase the number of hours that each capacity factor contributed in a full year. Equation 6.7 details the scaling process to turn the medoid or centroid, shown in Equation 6.5, into a scaled day. Where $\widetilde{P}_h^{x,i}$ is the scaled day:

$$\widetilde{P}_h^{x,i} = \{P_{1w_i}, P_{2w_i}, \ldots, P_{24w_i}\} \tag{4.8}$$

Equation 6.7 effectively extends the length of the day proportional to the amount of days in the respective cluster.

Finally, each of the scaled representative days were concatenated to create the series used for the calculations which required the capacity factors and the respective number of hours that each capacity factor contributed to the year. Equation 6.8 displays the total time series for series x, where each scaled medoid is concatenated to produce an approximated time series, $\widetilde{P}^x$:

$$\widetilde{P}^x = \left(\widetilde{P}_h^{x,1}, \widetilde{P}_h^{x,2}, \ldots, \widetilde{P}_h^{x,||N||}\right) \tag{4.9}$$

the total number of hours in the approximated time series, $\widetilde{P}^x$, is equal to the number of hours in a day multiplied by the number of days in a year, which gives the total number of hours in a year ($24 \times 365 = 8760$), as shown by Equation 6.9:

$$\sum_{w \in W} \sum_{t=1}^{T=24} (w_i t) = 24 \times 365 = 8760 \tag{4.10}$$

where $w \in W$ is the set of clusters.

4.9.2 Error Metrics

To measure the validity of our approximation using representative days and also compare the optimum number of days, or clusters, we used a technique similar to Poncelet *et al.* [44, 132]. We trialled the number of clusters against three different metrics: correlation (CE_{av}), normalised root mean squared error ($nRMSE$) and relative energy error (REE_{av}).

REE_{av} is the average value over all the considered time series $\widetilde{P}^x \in \widetilde{P}$ compared to the observed average value of the set $P^x \in P$. Where $P^x \in P$ are the observed time series and $\widetilde{P}^x \in \widetilde{P}$ are the scaled, approximated time series using representative days. REE_{av} is shown formally by Equation 6.10:

$$REE_{av} = \frac{\sum\limits_{P^x \in P} \left(\left| \frac{\sum_{t \in T} DC_{P_t^x} - \sum_{t \in T} \widetilde{DC}_{\widetilde{P}_t^x}}{\sum_{t \in T} DC_{P_t^x}} \right| \right)}{||P||} \tag{4.11}$$

where $DC_{P_t^x}$ is the duration curve for P^x and $DC_{\widetilde{P}_t^x}$ is the duration curve for $\widetilde{P}^x$. In this context, the duration curve can be constructed by sorting the capacity factor and electrical load data from high to low. The $x-$axis for the DC exhibits the proportion of time that each capacity factor represents. The approximation of the duration curve is represented in this text as $\widetilde{DC}_{\widetilde{P}^x}$.

$t \in T$ refers to a specific time step of the original time series. $\widetilde{DC}$ refers to the approximated duration curve for $\widetilde{P}^x$. Note that in this text $|\cdot|$ refers to the absolute value, and $||\cdot||$ refers to the cardinality of a set and $||P||$ refers to the total number of of considered time series.

Specifically, the sum of the observed values, P^x, and approximated values, $\widetilde{P}^x$, for all of the time series are summed. The proportional difference is found, which is summed for each of the different series, x, and divided by the number of series, to give REE_{av}.

Another requirement is for the distribution of load and capacity factors for the approximated series to correspond to the observed time series. It is crucial that we can account for both high and low levels of demand and capacity factor for IRES generation. This enables us to model for times where flexible generation capacity is required.

The distribution of values can be represented by the duration curve (*DC*) of the time series. Therefore, the average normalised root-mean-square error ($NRMSE_{av}$) between each *DC* is used as an additional metric. The $NRMSE_{av}$ is shown formally by Equation 6.11:

$$NRMSE_{av} = \frac{\sum_{P^x \in P} \left(\frac{\sqrt{\frac{1}{||T||} \cdot \sum_{t \in T} (DC_{P^x_t} - \widetilde{DC}_{\widetilde{P}^x_t})^2}}{max(DC_{P^x}) - min(DC_{P^x})} \right)}{||P||}. \tag{4.12}$$

Specifically, the difference between the approximated and observed duration curves for each time-step t is calculated. The average value is then taken of these differences. This average value is then normalised for the respective time series P^x. The average of these average normalised values for each time series are then taken to provide a single metric, $NRMSE_{av}$.

The final metric used is the correlation between the different time series. This is used due to the fact that wind and solar output influences the load within a single region, solar and wind output are correlated, as well as offshore and onshore wind levels within the UK. This is referred to as the average correlation error (CE_{av}) and shown formally by Equation 6.12:

$$CE_{av} = \frac{2}{||P|| \cdot (||P|| - 1)} \cdot \left(\sum_{p_i \in P} \sum_{p_j \in P, j > i} \left| corr_{p_i, p_j} - \widetilde{corr}_{p_i, p_j} \right| \right) \tag{4.13}$$

where $corr_{p1,p2}$ is the Pearson correlation coefficient between two time series $p_1, p_2 \in P$, shown by Equation 6.13. Here, $V_{p1,t}$ represents the value of time series p_1 at time step t:

$$corr_{p1,p2} = \frac{\sum_{t \in T} \left(\left(V_{p1,t} - \overline{V}_{p1} \right) \cdot \left(V_{p2,t} - \overline{V}_{p2} \right) \right)}{\sqrt{\sum_{t \in T} \left(V_{p1,t} - \overline{V}_{p1} \right)^2 \cdot \sum_{t \in T} \left(V_{p2,t} - \overline{V}_{p2} \right)^2}}. \tag{4.14}$$

4.9.3 Integrating higher temporal granularity

To integrate the additional temporal granularity of the model, extra time-steps were taken per year. The higher temporal granularity of the model enabled us to accurately model the hourly fluctuations in solar and wind which leads to more accurate expectations of the investment opportunities of these technologies [115, 72].

GenCos make bids at the beginning of every time-step and the Power Exchange matches demand with supply in merit-order dispatch using a uniform pricing market. An example of electricity mix in a single representative day is shown in Figure 6.1.

Figure 6.1 displays the high utilization of low marginal-cost generators such as nuclear, wind and photovoltaics. At hour 19, an increase in offshore wind leads to a direct decrease in CCGT. In contrast to this, a decrease in offshore and onshore between the hours of 8 and 12 lead to an increase in dispatch of coal and CCGT. One would expect this behaviour to prevent blackouts and meet demand at all times. This process has enabled us to more closely match fluctuations in IRES.

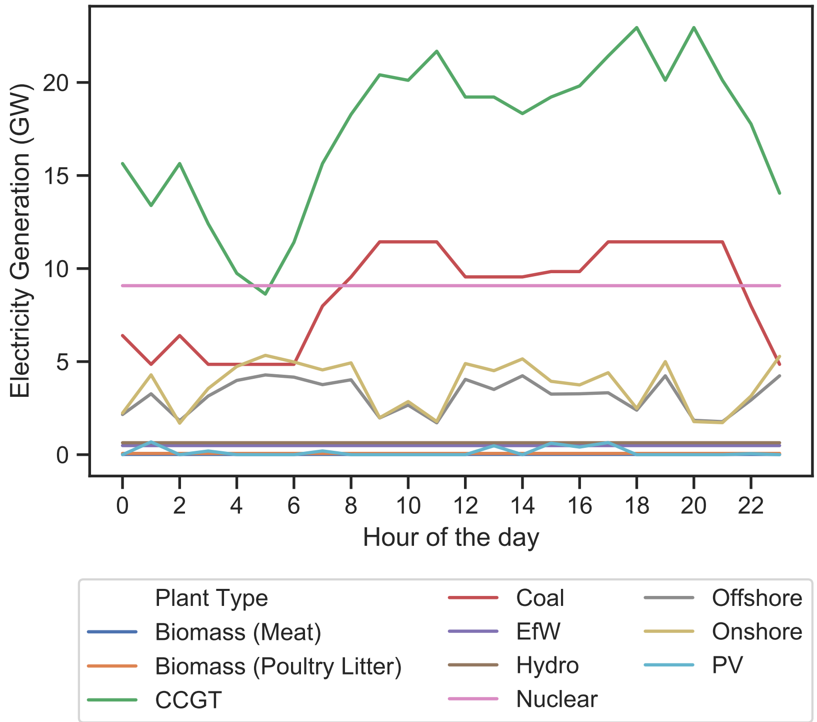


Fig. 4.8 Example of a single day of dispatched supply.

4.9.4 Genetic Algorithms

Genetic Algorithms (GAs) are a type of evolutionary algorithm which can be used for optimisation. We chose the genetic algorithm for this application due to its ability to find good solutions with a limited number of simulation runs, ability for parallel computation and its ability to find global optima. These characteristics are useful for our application, as a single simulation can take up to 36 hours.

In this section we detail the genetic algorithm used in this paper. Initially, a population P_0 is generated for generation 0. This population of individuals is used for the parameters to the simulation. The output of the simulations for each of the individuals are then evaluated. A subset of these individuals $C_{t+1} \subset P_t$ are chosen for mating. This subset is selected proportional to their fitness. With 'Fitter' individuals having a higher change of reproducing to create the offspring group C'_{t+1}. C'_{t+1} have characteristics dependent on the genetic operators: crossover and mutation. The genetic operators are an implementation decision [11].

Once the new population has been created, the new population P_{t+1} is created by merging individuals from C'_{t+1} and P_t. See Algorithm 2 for detailed pseudocode.

We used the DEAP evolutionary computation framework to create our genetic algorithm [55]. This framework gave us sufficient flexibility when designing our genetic algorithm. Specifically, it enabled us to persist the data of each generation after every iteration to allow us to verify and analyse our results in real-time.

Algorithm 1 Genetic algorithm [11]

$t = 0$
initialize P_t
evaluate structures in P_t
while termination condition not satisfied **do**
    $t = t + 1$
    select reproduction C_t from P_{t-1}
    recombine and mutate structures in C_t forming C'_t
    evaluate structures in C'_t
    select each individual for P_t from C'_t or P_{t-1}
end while

Parameters for Validation with Observed Data

The parameters chosen for the problem explained in Section 6.4.2 was a population size of 120, a crossover probability of 50%, a mutation probability of 20% and the parameters, m and c, as per Equation 6.3, were given the bounds of $[0.0, 0.004]$ and $[-30, 100]$ respectively.

The bounds for m and c were calculated to ensure a positive price duration curve, with a maximum price of 300 for $50,000$MW. The population size was chosen to ensure a wide range of solutions could be explored, whilst limiting compute time to $\sim$1 day per generation to allow for sufficient verification of the results. The crossover and mutation probabilities were chosen due to suggestions from the DEAP evolutionary computation framework [55].

Parameters for Long-Term Scenario Analysis

The parameters chosen for the genetic algorithm for the problem discussed in Section 6.4.3 are displayed here. The population size was 127, a crossover probability of 50%, a mutation probability of 20%. The parameters m_y, c_y were given the bounds $[0.0, 0.003]$ and $[-30, 50]$ respectively, whilst σ_m and σ_c were both given the bounds of $[0, 0.001]$.

The population size was slightly increased, and the bounds reduced when compared to the parameters for Section 6.5.4. This was to increase the likelihood of convergence to a global optima, which was more challenging to achieve due to the significantly higher number of parameters.

4.10 Results

Here we present the results of the problem formulation of Sections 6.4.2 and 6.4.3. Specifically, we compare the ability of our model to that of BEIS in the context of a historical validation between 2013 and 2018 of the UK electricity market. We also compare our ability to generate scenarios up to 2035 with that of BEIS.

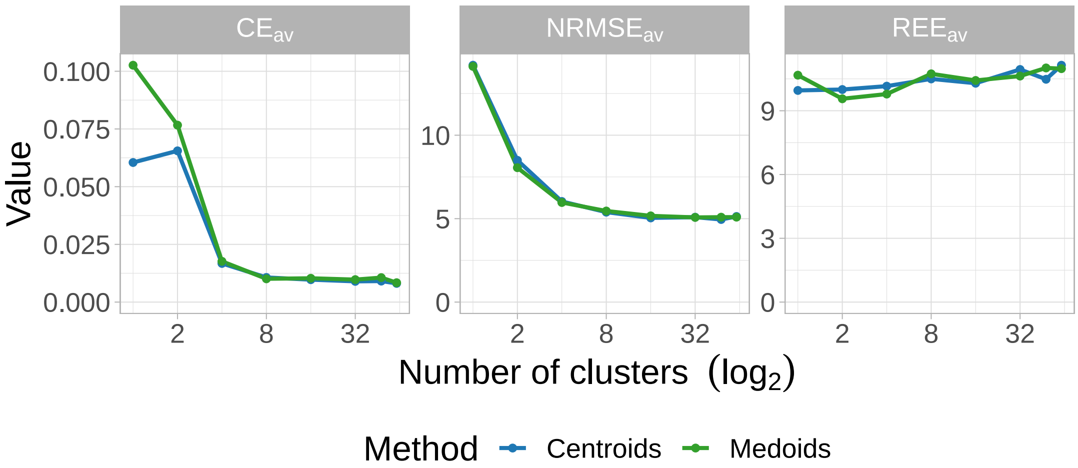


Fig. 4.9 Number of clusters compared to error metrics.

4.10.1 Selecting representative days

Figure 6.2 displays the error metrics versus number of clusters. Both CE_{av} and $NRMSE_{av}$ display similar behaviour, namely the error improves significantly from a single cluster to eight clusters for both centroids and medoids. For the number of clusters greater than eight there are diminishing returns. For REE_{av}, however, the error metric is best at a single cluster, and gets worse with the number of clusters.

We chose eight clusters as a compromise between accuracy of the three error metrics and compute time of the simulation. This is because, eight was the largest number of clusters that gave us the lowest score for CE_{av}, $NRMSE_{av}$ and REE_{av} without significantly increasing compute time. Whilst there was little significant difference between centroid and medoid, we chose to use the medoids due to the fact that the extreme high and low values would not be lost due to averaging [75].

4.10.2 Validation with Observed Data

Figure 6.3 displays the output of ElecSim under the validation scenario, BEIS' projections and the observed electricity mix between 2013 and 2018, as explained in Sub-Section 6.4.2.

The observed electricity mix changed significantly between 2013 and 2018. A continuous decrease of electricity production from coal throughout this period was observed. 2015 and 2016 saw a marked decrease of coal, which can be explained by the retirement of 3 major coal power plants. The decrease in coal between 2013 and 2016 was largely replaced by an increase in gas. After 2016, renewables play an increasingly large role in the electricity mix and displace gas.

Both ElecSim and BEIS were able to model the fundamental dynamics of this shift from coal to gas as well as the increase in renewables. Both models, however, underestimated the magnitude of the shift from coal to gas. This could be due to unmodelled behaviours such as consumer sentiment towards highly polluting coal plants, a prediction from industry that gas would become more economically attractive in the future or a reaction to The Energy Act 2013 which aimed to close a number of coal power stations over the following two decades [127].

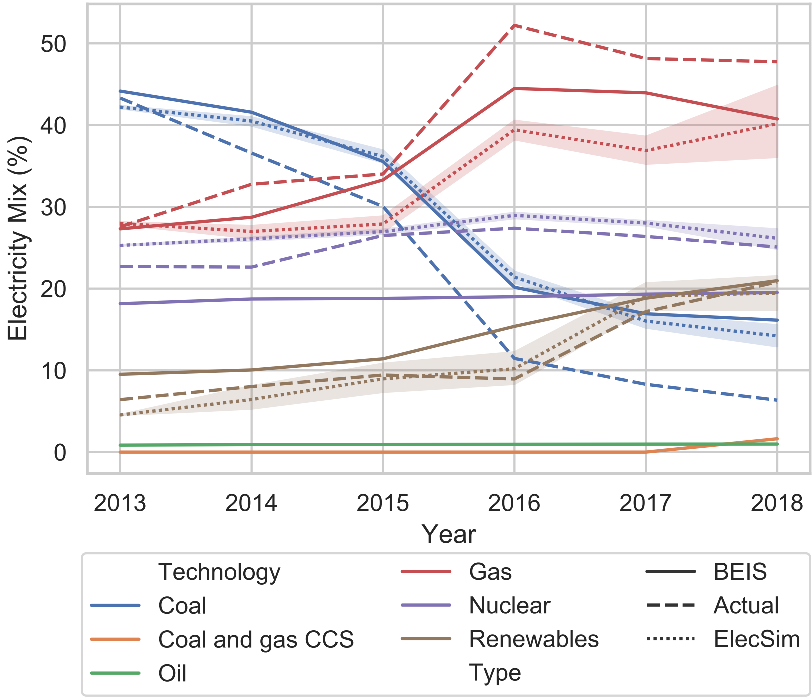


Fig. 4.10 Comparison of actual electricity mix vs. ElecSim vs. BEIS projections and taking three coal power plants out of service.

ElecSim was able to closely model the increase in renewables throughout the period in question, specifically predicting a dramatic increase in 2017. This is in contrast to BEIS who predicted that an increase in renewable energy would begin in 2016. However, both models were able to accurately predict the proportion of renewables in 2018.

ElecSim was able to better model the observed fluctuation in nuclear power in 2016. BEIS, on the other hand, projected a more consistent nuclear energy output. This small increase in nuclear power is likely due to the decrease in coal during that year. BEIS consistently underestimated the share of nuclear power.

We display the error metrics to evaluate our models 5 year projections in Table 6.1. Where MAE is mean absolute squared error, MASE is mean absolute scaled error and RMSE is root mean squared error.

We are able to improve the projections for all generation types when compared to the naive forecasting approach using ElecSim, as shown by the MASE. Where the naive approach is simply predicting the next time-step by using the last known time-step. In this case the last known time-step is the electricity mix percentage for each generation type in 2013.

Figure 6.4 displays the optimal predicted price duration curve (*PPDC*) found by the genetic algorithm. This price curve was used by the GenCos to achieve the results shown in Figure 6.3.

The yellow points show the simulated price duration curve for the first year of the simulation (2018). The red line (Simulated Fit (2018)) is a linear regression that approximates the simulated price duration curve (PDC (2018)). The blue line shows the price duration curve predicted (*PPDC*) by the GenCos to be representative of the expected prices over the lifetime of the plant.

The optimal predicted price duration curve (*PPDC*) closely matches the simulated fit in 2018, shown by Figure 6.4. However, the *PPDC* has a slightly higher peak price and lower baseload

Technology	MAE	MASE	RMSE
CCGT	9.007	0.701	10.805
Coal	8.739	0.423	10.167
Nuclear	1.69	0.694	2.002
Solar	0.624	0.419	1.019
Wind	1.406	0.361	1.498

Table 4.3 Error metrics for time series forecast from 2013 to 2018.

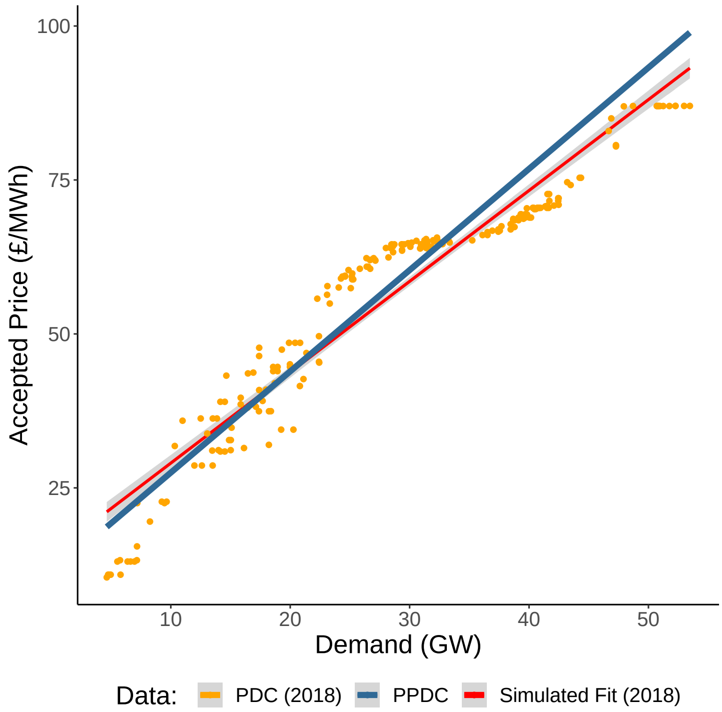


Fig. 4.11 Predicted price duration curve for investment for most accurate run against simulated run in 2018.

price. This could be due to the fact that there is a predicted increase in the number of renewables with a low SRMC. However, due to the intermittency of renewables such as solar and wind, higher peak prices are required to generate in times of low wind and solar irradiance at the earth's surface.

To generate Figure 6.5, we ran 40 scenarios with the *PPDC* to observe the final, simulated electricity mix. The error bars are computed based on a Normal distribution 95% confidence interval.

ElecSim was able to model the increase in renewables and stability of nuclear energy in this time. ElecSim was also able to model the transition from coal to gas, however, underestimated the magnitude of the transition. This was similar to the projections BEIS made in 2013 as previously discussed.

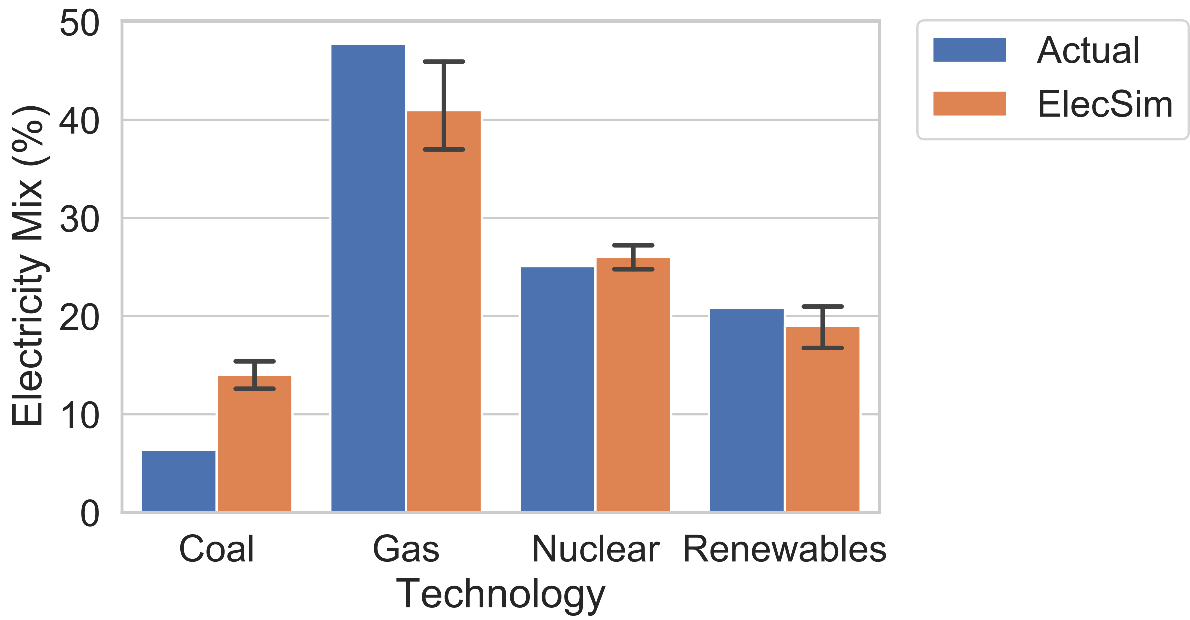


Fig. 4.12 Electricity generation mix simulated by ElecSim from 2013 to 2018 compared to observed electricity mix in 2018.

4.10.3 Long-Term Scenario Analysis

In this section we discuss the results of the analysis of the BEIS reference scenario explained in Section 6.4.3. Specifically, we created a scenario that mimicked that of BEIS in ElecSim and optimised a number of parameters using a genetic algorithm to match this scenario. Through this we are able to gain confidence in the underlying dynamics of ElecSim to simulate long-term behaviours. Further, this enables us to verify the likelihood of the scenario by analysing whether the parameters required to make such a scenario are realistic.

Figure 6.6 displays the electricity mix projected by both ElecSim and BEIS. To generate this image we ran 60 scenarios under the optimal collection of predicted price duration curves, nuclear subsidy and uncertainty in predicted price duration curves.

The optimal parameters were chosen by choosing the parameter set with the lowest mean error per electricity generation type and per year throughout the simulation, as shown by Equation 6.4.

Figure 6.7 displays the optimal predicted price duration curves (*PPDC*s) per year of the simulation, shown in blue. These are compared to the price duration curve simulated in 2018, as per Figure 6.4. The optimal nuclear subsidy, S_n, was found to be ∼£120, the optimal σ_m and σ_c were found to be 0 and ∼0.0006 respectively.

The BEIS scenario demonstrates a progressive increase in nuclear energy from 2025 to 2035, a consistent decrease in electricity produced by natural gas, an increase in renewables and decrease to almost 0% by 2026 of coal.

ElecSim is largely able to mimic the scenario by BEIS. A large increase in renewables is projected, followed by a decrease in natural gas.

A significant difference, however, is the step-change in nuclear power in 2033. This led to an almost equal reduction in natural gas during the same year. In contrast, BEIS project a continuously increasing share of nuclear.

We argue that the ElecSim projection of nuclear power is more realistic than that of BEIS due to the instantaneous nature of large nuclear power plants coming on-line.

Figure 6.7 exhibits the price curves required to generate the scenario show in Figure 6.6. The majority of the price curves are similar to the simulated price duration curve of 2018 (Simulated Fit (2018)). However, there are some price curves which are significantly higher and significantly lower than the predicted price curve of 2018. These cycles in predicted price duration curves may be explained by investment cycles typically exhibited in electricity markets [65].

In this context, investment cycles reflect a boom and bust cycle over long timescales. When electricity supply becomes tight relative to demand, prices rise to create an incentive to invest in new capacity. Price behaviour in competitive markets can lead to periods of several years of low prices (close to short-run marginal cost) [161].

As plants retire or demand increases, the market becomes tighter until average prices increase to a level above the threshold for investment in new power generators. At this point investors may race to bring new plants on-line to make the most out of the higher prices. Once adequate investments have been made, the market returns to a period of low prices and low investment until the next price spike [65].

The nuclear subsidy, S_n, of $\sim$£120 in 2018 prices is high compared to similar subsidies, but this may reflect the difficulty of nuclear competing with renewable technology with a short-run marginal cost that tends to £0.

The low values of σ_m and σ_c demonstrates that the expectation of prices does not necessarily have to differ significantly between GenCos. This may be due to the fact that GenCos have access to the same market information.

4.11 Conclusion

In this paper we have demonstrated that it is possible to use ABM to simulate liberalised electricity markets. Through validation we are able to show that our model, ElecSim, is able to accurately mimic the observed, real-life scenario in the UK between 2013 and 2018. This provides confidence in the underlying dynamics of ElecSim, especially as we are able to model the fundamental transition between coal and natural gas observed between 2013 and 2018 in the UK.

In addition to this, we were able to compare our long-term scenario to that of the UK Government, Department for Business, Energy & Industrial strategy. We show that we are able to mimic their reference scenario, however, demonstrate a more realistic increase in nuclear power. The parameters that were gained from optimisation show that the BEIS scenario is realistic, however a high nuclear subsidy may be required.

To improve the accuracy of our model, we used eight representative days of solar irradiance, offshore and onshore wind speed and demand to approximate an entire year. The particular days

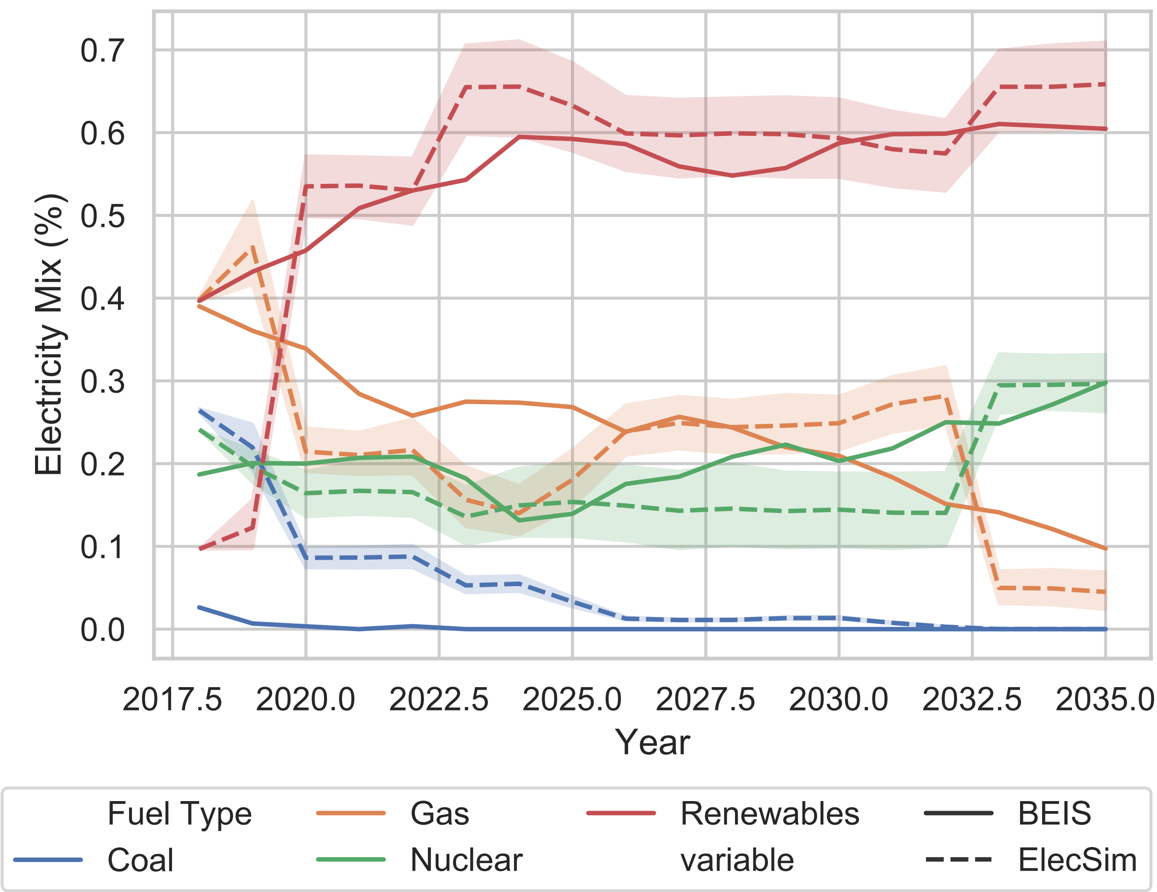


Fig. 4.13 Comparison of ElecSim and BEIS' reference scenario from 2018 to 2035.

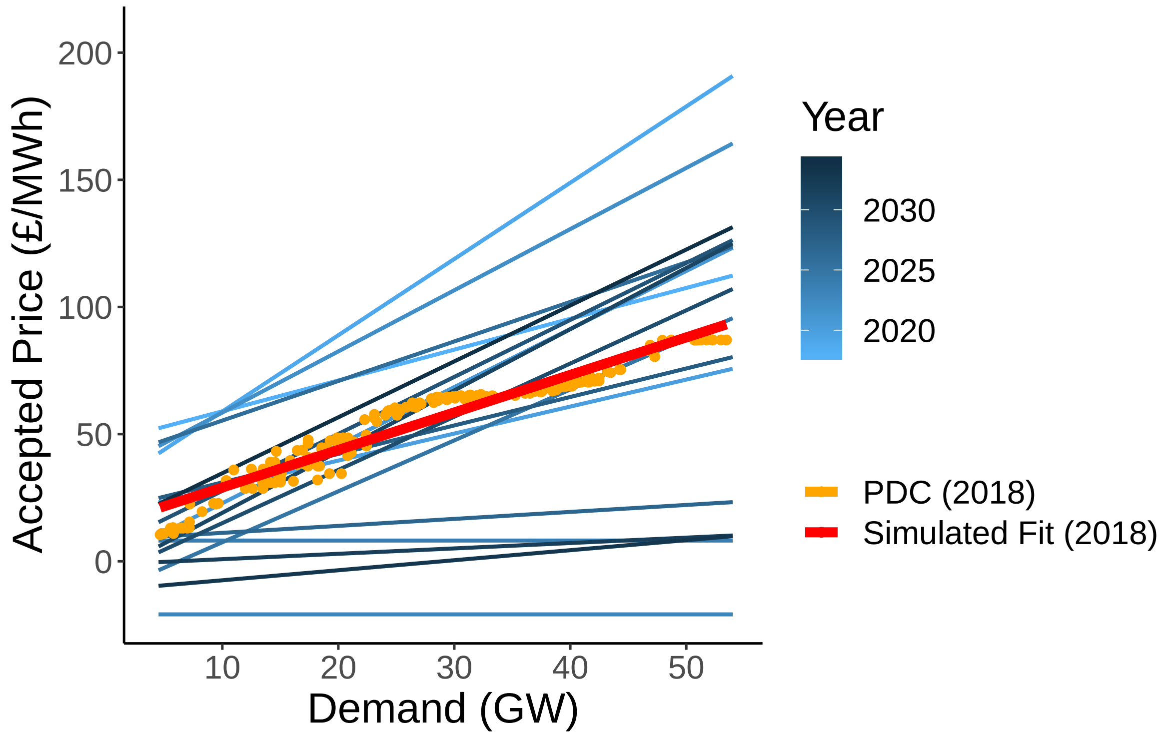


Fig. 4.14 Comparison between optimal price duration curves and simulated price duration curve in 2018.

were chosen using a k-means clustering technique, and selecting the medoids. This enabled us to accurately model the daily fluctuations of demand and renewable energy resources.

In future work we would like to evaluate further scenarios to provide advice to stakeholders, integrate multi-agent reinforcement learning techniques to better model agents in both investment and bidding strategies as well as model different countries. Further work could be to make predicted price duration curves endogenous to the model, however, this could require scenario analysis by each of the GenCos each time they wanted to make an investment.

In addition to this, a method of dealing with the non-validatable nature of electricity markets, as per the definition of Hodges *et al.* is to vary input parameters over many simulations and look for general trends [80]. This could be achieved using ElecSim through the analysis of a reference case, and a limited set of scenarios which include the most important uncertainties in the model structure, parameters, and data, i.e. alternative scenarios which have both high plausibility and major impacts on the outcomes.

Chapter 5

Electricity demand prediction

5.1 Full paper

5.2 Introduction

The energy markets have undergone significant changes in recent years. The liberalisation of the energy industry, technological advancements and policy changes have had a number of effects [158] including, a rise in both competition [146], and quantity of data collected [31], as well as a requirement to integrate large amounts of intermittent renewable resources [68, 99, 25].

The need for accurate load forecasting is essential for control and planning of electricity generation in electrical grids due to the fact that supply must meet demand [114]. Short-term electricity demand forecasting has become increasingly important due to the introduction of competitive energy markets. Accurate estimates of demand are required so that the correct amount of electricity is purchased on the wholesale market [45]. Electricity is unique to other commodities in that it must be either consumed the moment that it is generated or stored. The difficulties in storing electricity arise from high installation and maintenance costs, inefficiencies and low capacity [133]. It is therefore important to match demand to supply and thus regulate frequency. Failure to accurately forecast electricity demand can lead to financial loss and/or system-wide blackouts [77].

The introduction of smart meters in many countries (USA, Europe, Canada and South Korea) has led to an influx of high granularity electricity consumption data that can be used for load forecasting [43]. 800 million smart meters are projected to be installed worldwide by 2020 [152]. Smart meters are digital devices that measure electricity consumption of individual households at regular intervals (intervals of an hour or less) and offer two-way communication between the meter and utility company. Smart meters aid customers to understand precisely how much electricity they consume at different time intervals, and enable dynamic pricing [5]. Dynamic pricing allows utilities to charge varying prices at different times, for instance, charging a higher price when costly generation sources are used in times of peak demand, and lower prices at night time or weekends when demand is low [112, 94].

This paper explores short term load-forecasting at an interval of 30 minutes ahead and clusters similar users based on their electricity usage. A variety of different forecasting techniques were evaluated such as Random Forests [Tin Kam Ho], Long-Short Term Memory neural networks (LSTM) [79], Multilayer Perceptron neural networks [157] and Support Vector Regression (SVR) [46].

Random Forests are an ensemble based learning method for classification and regression, and are made up of many decision trees. LSTM networks are recurrent neural networks which remember values over arbitrary time intervals. Multilayer Perceptrons are a popular type of neural network which are made up of a minimum of three layers and can be used to make non-linear predictions. SVRs are supervised learning models which analyze data used for regression analysis.

To improve forecasting results, the clustering of smart meter data was evaluated. The technique used for this was k-means clustering. An average 24-hour electricity load profile was calculated, and the result used for clustering. The clustered sub-system is then aggregated and separate models trained on this aggregate. The yearly, weekly and daily periodicity of electricity load is accounted for by input variables into the models. Once forecasts for each cluster are made using the individual models, the results are aggregated for the final predictions. These predictions are compared to the actual results and the accuracy measured using mean absolute percentage error (MAPE).

This paper contributes a technique to forecasting smart-meter data through the use of clustering using k-means and different learning algorithms. It provides researchers and utilities with methods to maximise forecasting accuracy through the selection of machine learning and clustering algorithms.

This paper is structured as follows: in Section 2 we introduce the state of the art in load forecasting. The methods used in this paper are explored in Section 3. The experiments and their evaluation are discussed in Section 4. The results are discussed in Section 5. In Section 6 we conclude and consider future directions for this work.

5.3 Related Work

The forecasting of aggregated and clustered electricity demand has been the focus of a considerable amount of research in recent years. The research can generally be classified into two classes, Artificial Intelligence (AI) techniques [105, 122, 135] and classical time series approaches [**?** 86, 125]. For the purposes of our paper we have evaluated artificial intelligence techniques.

Singh *et al.* [145] produced a review of load forecasting techniques and methodologies and reported that hybrid methods, which combine two or more different techniques, are gaining traction, as well as soft computing approaches (AI) such as genetic algorithms. Our paper presents a hybrid method which combines k-means clustering with multiple different learning algorithms.

5.3.1 Artificial Intelligence Techniques

Dillon *et al.* presented a neural network for short term load forecasting. Their neural network consisted of three-layers and used adaptive learning for training [45]. They proposed the use of weather information to augment their electricity load data. They found better results with the adaptive neural network than with a linear model, or non-adaptive neural network. In contrast to Dillon our paper focuses on a non-adaptive neural network and does not take into account weather information.

Chen *et al.* used an Artificial Neural Network to predict electricity demand of three substations in Taiwan. They integrated temperature data and reported that the best results when forecasting residential and commercial substations were during the week due to the influence of weather [28]. In contrast to the work done by Chen *et al.*, we focus on client-side prediction using smart meter data as opposed to substation data. We were, therefore, able to cluster the data based on load profile, as opposed to geographical location.

5.3.2 Time Series Approach

Al-Musaylh *et al.* proposed the use of SVR, an autoregressive integrated moving average (ARIMA) model and a multivariate adaptive regression spline (MARS) in their short term electricity demand forecasting system [8]. They found that for a half, and one-hour forecasting horizons, that the MARS model outperformed both the ARIMA and SVR.

Taylor evaluates different statistical methods including ARIMA, an adaptation of Holt-Winters' exponential smoothing [83], and an exponential smoothing method which focuses on the evolution of the intra-day cycle [73]. He found that the double seasonal adaptation of the Holt-Winters' exponential smoothing method was the most accurate method for short lead times between 10 and 30 minutes.

In contrast to Taylor, Fard *et al.* proposed a novel hybrid forecasting method based on both artificial intelligence and classical time series approaches. They utilised the wavelet transform, ARIMA and ANNs for short term load forecasting [49]. The ARIMA model is created by finding the appropriate order using the Akaike information criterion [7]. The ARIMA model models the linear component of the load time series, and the residuals contain the non-linear components. These residuals are then decomposed by the discrete wavelet transform into its sub-frequencies. ANNs are then applied to these sub-frequencies and the outputs of both the ANN and ARIMA models are summed to make the final prediction. They found that this hybrid technique outperformed traditional methods. Our paper, however, does not integrate artificial intelligence and classical time series techniques.

5.3.3 Clustering

Multiple techniques have been proposed for the clustering of electricity load data prior to forecasting. Both Shu and Luonan, and Nagi *et al.* propose a hybrid approach in which self-organizing maps are used to cluster the data, and Support Vector Regression is used to make predictions [144, 122]. This technique proved robust for different data types, and was able to tackle the non-stationarity of the data. Shu showed that this hybrid approach out-performed a single SVR technique, whilst Nagi showed superior results to a traditional ANN system. In contrast to both Nagi *et al.* and Shu and Luonan our paper utilises k-means as the clustering algorithm

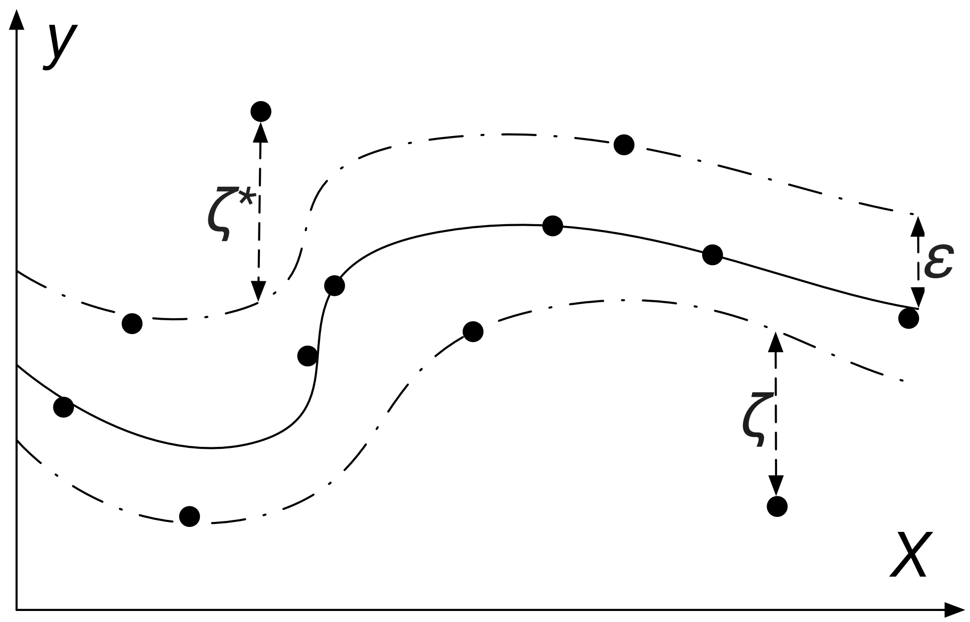


Fig. 5.1 A three layer feed forward neural network.

Similarly to us, Quilumba *et al.* proposed a k-means clustering algorithm to cluster similar load profiles [135]. They found that clustering does improve accuracy, and found an optimum number of 3 clusters.

Wijaya *et al.* demonstrated that implementing clusters improved load-forecasting accuracy up to a certain level [163]. Whilst, a study by Ilić *et al.* showed that increasing the number of clusters did not improve accuracy [91].

5.4 Time series forecasting methods

In this section, the basic principles behind ANNs, Support Vector Regression and Random Forests are discussed.

5.4.1 Artificial Neural Network

Artificial Neural Networks are a type of model which allow for non-linear relationships to be modeled between the input and output data [7]. A popular neural network is a feed forward multilayer network. Fig. 5.13 shows a three layer feed forward neural network with a single output unit, k hidden units, n input units. w_{ij} is the connection weight from the ith input unit to the jth hidden unit, and T_j is the connecting weight from the jth hidden unit to the output unit [128]. These weights transform the input variables in the first layer to the output variable in the final layer based upon the training data.

Typically, a dataset is split into three sections, the test set, training set and validation set. The training set is used to find the connection weights of the network, whilst the test set is used to determine the accuracy of the models. The validation set allows for an unbiased evaluation of the model whilst tuning the hyperparameters, and can avoid overfitting by stopping training if the error begins to increase.

For a univariate time series forecasting problem, suppose we have N observations $y_1, y_2, \dots, y_N$ in the training set,

$$y_{N+1}, y_{N+2}, \dots, y_{N+m} \tag{5.1}$$

in the test set and we are required to predict m periods ahead [128].

The training patterns are as follows:

$$y_{p+m} = f(y_p, y_{p-1}, \dots, y_1) \tag{5.2}$$
$$y_{p+m+1} = f(y_{p+1}, y_p, \dots, y_2) \tag{5.3}$$
$$\vdots$$
$$y_N = f(y_{N-m}, y_{N-m-1}, \dots, y_{N-m-p+1}) \tag{5.4}$$

where f is the function made up of weights and activation functions in the trained neural network.

The m testing patterns are

$$y_{N+1} = f(y_{N+1-m}, y_{N-m}, \dots, y_{N-m-p+2}) \tag{5.5}$$
$$y_{N+2} = f(y_{N+2-m}, y_{N-m+1}, \dots, y_{N-m-p+3}) \tag{5.6}$$
$$\vdots$$
$$y_{N+m} = f(y_N, y_{N-1}, \dots, y_{N-p+1}) \tag{5.7}$$

The training objective is to minimize the overall predictive error means (SSE) by adjusting the connection weights. For this network structure the SSE can be written as:

$$SSE = \sum_{i=p+m}^{N} (y_i - \hat{y}_i) \tag{5.8}$$

where $\hat{y}_i$ is the output from the network. The number of input nodes corresponds to the number of lagged observations. Having too few or too many input nodes can affect the predictive ability of the neural network [128].

5.4.2 Support Vector Regression

A Support Vector Regression model maps input data, x, into a higher-dimensional feature space non-linearly. Given the input data:

$$(x_1, y_1), \dots, (x_i, y_i), \dots, (x_n, y_n) \tag{5.9}$$

where x_i is the input, and y_i is the output value of x_i. Support Vector Regression solves an optimization problem [144, 27]

$$\min_{\omega, b, \xi, \xi^*} \frac{1}{2}\omega^T \omega + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \tag{5.10}$$

subject to

$$\begin{aligned} y_i - (\omega^T \phi(x_i) + b) &\leq \varepsilon + \xi_i^*, \\ (\omega^T \phi(x_i) + b) - y_i &\leq \varepsilon + \xi_i, \\ \xi_i, \xi_i^* &\geq 0, i = 1, \ldots, n \end{aligned} \tag{5.11}$$

x_i is mapped to a higher dimensional space using the function ϕ. The ε-insensitive tube $(\omega^T \phi(x_i) + b) - y_i \leq \varepsilon$ is a range shown in Figure 5.2 in which errors are permitted. ξ_i and ξ_i^* are slack variables which allow errors for data points which fall outside of ε. This enables the optimisation to take into account the fact that data does not always fall within the ε range [147].

The constant $C > 0$ determines the trade-off between the flatness of the support vector function. ω is the model fit by the SVR. The parameters which control regression quality are the cost of error C, the width of the tube ε, and the mapping function ϕ [144, 27].

Figure 5.2 demonstrates this principle, where only data points which fall outside of the ε-insensitive tube are penalised in the cost function

The constraints shown in (5.22) imply that most of the data x_i falls within the tube ε. By minimizing the training error $C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$, and the regularisation term $\frac{1}{2}\omega^T \omega$, under-fitting and over-fitting are avoided.

Due to the fact that ϕ maps x_i to a high or infinite dimensional space, ω can be solved, subject to the constraints in (5.22), by Lagrangian optimisation. Thus the dual problem is solved [144, 27, 147]:

$$\min_{\alpha\alpha^*} \frac{1}{2}(\alpha - \alpha^*)^T Q(\alpha - \alpha^*) + \varepsilon \sum_{i=1}^{n} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \tag{5.12}$$

subject to

$$\begin{aligned} \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) &= 0, \\ 0 \leq \alpha_i, \alpha_i^* &\leq C, i = 1, \ldots, n \end{aligned} \tag{5.13}$$

where $Q_{ij} = \phi(x_i)^T \phi(x_j)$, a_i and a_i^* are Lagrange multipliers. However due to the large number of elements in $\phi(x)$ we apply a "kernel trick" to do the mapping implicitly. Now, the optimisation can be calculated using solely dot products. An example of the linear function kernel is listed below, which is used in this paper:

$$K(x, y) = x^T y. \tag{5.14}$$

5.4.3 Decision Tree

A decision tree is a statistical model used for either classification or regression [22]. Load forecasting is a regression problem, and therefore regression trees are used in this paper.

Decision trees recursively partition data into nodes with different labels until the termination criteria is met. This is typically initiated when it is not possible to have children nodes with different output labels. The terminal nodes are known as leaves and they represent the different outputs.

5.4.4 Random Forest

A Random Forest is an ensemble method that combines the predictions of many decision trees [21]. Each decision tree is fit by a random sample, with replacement, of the training data. The final decision of the Random Forest is decided by a majority case wins vote on each of the decision trees.

5.4.5 MAPE

For the measure of prediction accuracy, this paper adopts mean absolute percentage error (MAPE). The formula is as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{5.15}$$

where y_i is the actual value, $\hat{y}_i$ is the forecast value and n is the number of points forecast [111].

5.4.6 Root Mean Squared Error

The root mean squared error (RMSE) is a measure between the values predicted by a model and the observed values. The RMSE is the sample standard deviation of the differences between the predicted and observed values.

The RMSE is defined as follows:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_i - y_i)^2)}{n}} \tag{5.16}$$

where $\hat{y}_i$ are the predicted values, y_i are the observed values, and n is the number of observations.

5.5 Methodology

In this section, we will present the methodology for the approach taken in this paper.

The work in this paper was run on a MacBook Pro with a quad-core 3.1GHz Intel Core i7 processor with 16 GB 1867 MHz DDR3 of RAM and a 500GB solid state drive (SSD).

5.5.1 Data Collection

Smart meter data obtained from the Irish Social Science Data Archive (ISSDA) on the 28th of September 2017 was used in this study [52]. The Commission for Energy Regulation released a public dataset of anonymised smart meter data from the "*Electricity Smart Metering Customer Behaviour Trials*" [3]. This dataset is made up of over 5000 Irish homes and businesses and is sampled at 30-minute intervals.

The data was recorded between the 14th July 2009 and 31st December 2010, providing 17 months worth of data. For the purposes of cross-validation the data was split into two partitions, the training set and the testing set. The training set made up the first 11 months of data and was used to parametrise the models, whereas the test set is made up of the remaining 6 months of data. This split was chosen to balance the amount of training data with the test data and to give the models a chance to learn the periodicity inherent in a one year period of electricity load. The test set was used for evaluation of the models proposed. Due to the long training times for these algorithms, we worked with a sub-sample of 709 individual Irish homes from the whole dataset. However, we believe that our results would hold over the full dataset.

Figure 5.3 demonstrates the electricity consumption profile of a single week for a single user. Whilst it can be seen that electricity usage changes significantly between days, a pattern of behaviour is exhibited. There is a large peak displayed each day in the evening, as well as a peak earlier during the day. It can, therefore, be assumed that this customer has some form of habitual behavioural pattern.

Figure 5.4 shows eight different residential customer load profiles on the 22nd June 2009. It can be seen that the daily load profile changes between each customer. The consumers use varying quantities of electricity and at different times.

These figures display that electricity consumption changes per person, per day. To capture this variability between customer types these customers are clustered and then aggregated. Each of the different aggregated electricity consumptions should provide a less stochastic load profile, and therefore increase the accuracy of the models.

5.5.2 Clustering

We propose that clustering similar customer load profiles and aggregating each cluster's electricity consumption improves the accuracy of the models.

Figure 5.10 displays four different customers with similar load profiles. Each of the users display a strong peak in electricity consumption during the evening and less consumption during the day. These customers may potentially be clustered together by the k-means clustering algorithm.

To cluster the load profiles different options were considered. Hierarchical clustering using metrics such as Euclidean and wavelet distance metrics were evaluated [119], as was k-means [**?**]. K-means demonstrated to be the most robust and best-performing clustering algorithm, and thus was chosen for use in this paper.

To select the optimum number of clusters (k) cross-validation was explored. This allowed us to compare the results of each of the models and select k with the highest MAPE accuracy.

The cross-validation method proposed, worked by trying a different number of clusters per model, and testing for the resulting MAPE. The optimum number of clusters with a low MAPE is then chosen. In this paper we varied k between 1 and 7, this range was chosen due to the fact that the error did not vary greatly past seven clusters. We fit multiple models per cluster and predicted 6 months of electricity consumption.

With k-means clustering, it is possible that with the same initialization number of clusters, different clusters are formed. This is due to the algorithm converging at a local minima. To overcome local minima the k-means algorithm is run multiple times and the partition with the smallest squared error is chosen [96]. In our case, the k-means clustering algorithm is run 1000 times to reduce the chance of finding a local minima.

The clustering technique utilised in our paper was a scaled input approach. The daily load profile was averaged for each customer based on each day of the training data. The data was then scaled so that households of different sizes, but with similar usage profiles were clustered together. This data, which is made up of a *m-by-n* matrix, where m is equal to the total number of meters and n is equal to 48 (two readings for each hour in the day).

To find the optimum number of clusters it is recommended that the user selects a value of k that is high enough that distinct average load profiles are displayed, however, not so high that well-clustered customers are split. By doing this, the stochasticity of the load profiles in each of the clusters will be reduced, and thus lead to the best results.

5.5.3 Aggregating Demand

Once each customer is assigned to their respective cluster, the total electricity consumed per cluster is aggregated. This is achieved by summing the electricity consumed at each time interval per cluster. This creates a partial system load. A different model is trained on each of the different partial system loads, and the resultant forecasts are aggregated to generate the total system load forecast. The total system load forecast is then used to evaluate the accuracy of each of the different models using MAPE.

Random Forests, Support Vector Regression, Multilayer Perceptron neural networks and Long-Short Term Memory neural networks were evaluated, and a comparison between the different models were made.

These models were chosen due to their ability to model multivariate non-linear relationships. They are data-driven methods and therefore suited to this type of problem.

5.5.4 Feature Selection

Each component of the training data is known as a feature. Features encode information from the data that may be useful in predicting electricity consumption.

Calendar Attributes

Due to the daily, weekly and annual periodicity of the electricity consumption daily calendar attributes may be useful to model the problem. The calendar attributes included are as follows:

- Hour of day
- Day of the month
- Day of the week

- Month
- Public holidays

These attributes enable the daily, weekly and annual periodicity to be taken into account by the model.

It is noted that electricity consumption changes on a public holiday such as Christmas or New Year's Eve. It is therefore proposed that public holidays in Ireland are input into the model as features.

For testing purposes, two sets of models for Random Forests, Multilayer Perceptrons and Support Vector Regression were fit. One set omitted these calendar attributes whilst the other didn't. This is done to evaluate the importance of periodicity in electricity consumption prediction.

Time Series Data

As well as the calendar attributes it is important to consider the historical load demand. This allows the time-series element to be modelled.

To do this, a lagged input of the previous 3 hours, the equivalent three hours from the previous day, and the equivalent 3 hours from the previous week were used. For example, to predict the electricity consumed on the 21st December 2010 at 12:00 pm the electricity between 9:00 pm and 11:30 pm on the 21st of December are used as inputs, as are the times between 9:00 pm and 12:00 pm on the 20th and 14th of December.

Long-Short Term Memory neural networks remember values over arbitrary time intervals. They can remember short-term memory over a long period of time, for this reason, 5 lagged inputs of the previous two and a half hours were used as features to the Long-Short Term Memory network.

Data Representation

Once useful information is selected we must encode the data for input into the models. To encode the day of the week seven binaries are utilised. Six of the binaries are for Monday through to Saturday. When all six binaries are equal to zero Sunday is encoded. A single binary for public holidays is included. Eleven binaries are used for month of the year, with the first eleven representing January to November, with December represented by all zeros in the calendar binaries. The current hour and date are input using a numerical attribute. The lagged data inputs, such as previous hour's electricity usage are also input using a numerical attribute for each entry, totaling 20 attributes (six half hourly entries for each 3 hour period multiplied by three days plus 2 entries for the time to be predicted on the previous day and week). Table 5.1 displays these features.

5.5.5 Experiments

This section explores the different methods used to select the model parameters, and the tests to evaluate our models. Once the parameters were chosen the models were trained on the different

Table 5.1 List of Input Data for Models

Input	Variable	Detail description
1	Hour	Single numeric input representing hour of the day
2	Day of month	Single numeric input representing day of the month
3-9	Day of week	Six binary digits representing calendar information regarding day of the week
10-21	Month	Eleven binary digits representing calendar information regarding month
22-42	Lagged inputs	Twenty numeric inputs representing lagged inputs of previous 3 hours, previous 3 hours of previous day including hour to be predicted, and previous 3 hours of previous week including hour to be predicted
43	Holiday	One binary digit representing whether the day was a public holiday

Kernel Type	Kernel Parameters	RMSE
Linear	No values	0.02103
RBF	C=2, $\gamma = 0.016$	0.0245
Polynomial	C=2, $d = 2, r = 2$	0.0315

Table 5.2 Prediction Accuracy Based on Type of Kernel

clusters of residential customers. Each model was run five times to explore the variance of the results.

Support Vector Regression

To implement a Support Vector Regression model a variety of parameters must be chosen. These parameters influence the performance of the model. The parameters were evaluated using cross-validation. To do this, the data was split 75% into training data, and the remaining 25% into test data. This split was chosen to balance the trade-off between having enough training data so that the model can accurately learn the underlying form of the data, but also to have enough data to test each model.

To choose the optimum support vector machine kernel cross-validation was also used. Again, with 75% acting as the training data and 25% as the test. The kernels compared were polynomial, radial basis function (RBF) and the linear kernel [24, 153]. These were chosen due to their popularity, support and relative speed of computation.

The parameter values selected are shown in Table **??**. These parameter values were chosen from the results of cross-validation for each of the different kernels. From the cross-validation, the linear kernel was found to be the best performing. For this reason, the linear kernel was utilised for prediction of electricity consumption in this paper.

Random Forest

To initialize the Random Forest algorithm with the number of variables randomly sampled as candidates at each split, cross-validation was used. Once again, 75% of the data was used for training and the remaining 25% for testing due to the trade-off between training and testing.

Figure 5.6 shows the results of tuning the parameter of the number of variables randomly sampled as candidates at each split. The optimum number was found to be 23. Either side of this value the RMSE increases. Therefore the value 23 was selected to be the number of variables randomly sampled as candidates at each split in the Random Forest model. It is proposed that the value 23 was found to be optimum due to the 20 lagged inputs, as this data is crucial for the Random Forest to learn the underlying nature of electricity load.

Multilayer Perceptron

A feed-forward Multilayer Perceptron is a common neural network architecture used for the prediction of time series data, which has comparable, and occasionally better results than statistical models [76].

The first step when designing a Multilayer Perceptron neural network is to design the architecture. For this case, the number of input neurons is set to 41 (see Table 5.1). Once an input for each neuron is entered, the output layer must be designed. Due to the fact that we are forecasting only one time step ahead (30 minutes ahead) one output neuron is required.

The next step is to design the architecture of the hidden layers. To accomplish this, cross-validation is utilised as per the previous models. A maximum of 3 hidden layers were tested and the results analysed. A similar method to Fan *et al.* was evaluated to choose the number of neurons and hidden layers, a technique known as the Levenberg-Marquardt technique [48]. The Levenberg-Marquardt is a technique suitable for training medium-sized Artificial Neural Networks with a low mean-squared error.

The fundamental rule is to select the minimum number of neurons in the hidden layer so as to capture the complexity of the model, but not too many as to introduce over-fitting, which results in a loss in generalization of the algorithm.

The method begins by choosing a small number of neurons and gradually increasing the number each time the model is trained and the forecast error obtained. The forecast error is monitored until an optimum value is found, to which no further improvement is noted. Once the optimum number of neurons in the layer is obtained an additional layer is added, and the same technique is used.

Using this technique an optimal architecture with three layers is obtained. The first layer contained two neurons, the second contained five, and the third contained four.

LSTM

To initialize the LSTM, cross-validation was used to select the number of stacked layers and memory units. Similarly to the technique used for the Multilayer Perceptron, the Levenberg-Marquardt was used. The optimum number of layers was found to be 2, with a total of 50 memory units. Different combinations of layers and memory units displayed worse results.

5.6 Results

To test the accuracy of the trained model the data was split into a training and test set. The data between the 14th of July 2009 and the 15th of June 2010 was used as the training data, whilst the data between the 15th of June 2010 and 31st of December 2010 was used for testing purposes. The test set is separate from the training set and not used during training.

28 independent forecasting models are constructed for each of the Random Forests, Support Vector Regression, LSTMs and Multilayer Perceptron neural networks for each of the groups with k varying from 1 to 7. This was done to determine the optimal number of clusters. Each of the 28 models are trained independently, five times each so that the standard deviation results of MAPE for each cluster could be displayed. We evaluated the MAPE of the overall prediction.

Figure 5.11 displays the accuracy of the models trained at different numbers of clusters (k). The results demonstrate that introducing clusters to group similar customers improve results in all cases. The optimum value for k for Random Forests, Support Vector Regression and neural networks was shown to be four for our dataset. After this, the accuracy diminishes slightly. The error bars shown in Figure 5.11 show a small variance in MAPE in SVRs, ANNs and Random Forests. However, the MAPE of the LSTMs seem to vary by up to 11% in the five models run.

Figure 5.9 demonstrates the impact of using calendar attributes such as month, day of the month, and day of the week on prediction accuracy. The results show an increase in prediction accuracy of 6% for neural networks, 4% for Random Forests and 1% for support vector regression when taking into account these variables. It is proposed that the ability for the models to take into account the cyclic yearly, monthly and weekly behaviour improves the results.

Figure 5.8 shows the average load profiles of different clusters when $k = 4$. It is proposed that the optimum number of clusters is four due to the distinct load profiles that can be seen in Figure 5.8. The four different distinct patterns seen are high night time use in cluster 1, a typical residential load profile is shown in cluster 2, a spread of usage in cluster 3, and high daytime usage in cluster 4. At $k = 3$ these distinct patterns are not adequately clustered, and at $k = 5$ one of the distinct clusters are split, leading to an increase in stochasticity.

It is true that the optimum number of clusters will vary for different datasets. Whilst residential smart meter datasets may be similar, it is entirely possible that different geographies display different usage characteristics based on factors such as culture, temperature and economical reasons. It is therefore important to choose an optimal number of clusters for each dataset.

The results demonstrate that SVR, Random Forests and the Multilayer Perceptrons have a similar overall accuracy. The LSTM shows a similar pattern in increasing accuracy with number of clusters. However, the Random Forest seems to outperform each of the models at every point. This may be due, in part, to the internal operation of the Random Forest which undertakes its own cross-validation using out-of-bag samples and only having a few tuning parameters.

It has been shown that neural networks, SVR and Random Forests all perform within an adequate range of predicting electricity consumption. Whilst LSTMs perform poorly. This may be due to the features given to the LSTM which only had previous two and a half hours of data as input.

However, it is well known that the best machine learning technique for predicting energy consumption cannot be chosen *a priori*. Therefore it is necessary to compare different techniques to find the best solution to a particular regression problem [6].

For this work, the training time was tested by timing how long the models would be fit to create one cluster (single model trained on the training set). The Support Vector Regression took much less time than all of the other methods, whereas the LSTM took the longest. The Artificial Neural Network required 9 minutes and 5 seconds to run. The Support Vector Regression model required 3 minutes and 32 seconds to run. The Random Forest, on the same data, required 9 minutes and 44 seconds to run, whilst the LSTM took 12 minutes 55 seconds.

5.7 Conclusion

The availability of high granularity data produced by the smart grid enables network operators to gain greater insights into their customer behaviour and electricity usage. This enables them to improve customer experience, utility operations and power management. We demonstrated that implementing the k-means clustering algorithm to group similar customers improved the accuracy of every one of the different models tested. Distinct models were trained for each of the clusters and the individual forecasts aggregated for the total aggregated forecast. It was found that Random Forests outperformed the other models at all levels of clustering and that the optimum number of clusters was 4. Whilst the dataset used focused on residential data it is expected that applying a similar clustering technique on commercial properties would have a similar effect.

In future work, we will look into the features that best aid in the forecasting of electricity consumption, try a wider variety of models in an ensemble manner and try different clustering techniques such as self-organizing maps (SOM) to obtain better accuracy measures. We will also compare different prediction error measures.

To utilize more of the data and increase the number of models trained these results could be run in parallel and on the cloud in future.

5.8 Note paper

5.9 Introduction

The energy markets have undergone significant changes in recent years. The liberalisation of the energy industry, technological advancements and policy changes have had numerous effects [158]. These include a rise in both competition and data [146, 31].

Accurate load forecasting is essential for control and planning of electricity generation in electrical grids as supply must meet demand [114]. Accurate estimates of demand are required so that the correct amount of electricity is purchased on the wholesale market [45]. Failure to accurately forecast electricity demand can lead to financial loss or system-wide blackouts [77].

The introduction of smart meters in many countries (USA, Europe and South Korea) has led to an influx of high granularity electricity consumption data that can be used for load forecasting [43]. 800 million smart meters are projected to be installed worldwide by 2020 [152].

This paper explores short-term load-forecasting at an interval of 30 minutes ahead and clusters similar users based on their electricity consumption. A variety of different forecasting techniques were evaluated such as Random Forests [Tin Kam Ho], Long Short-Term Memory Neural Networks (LSTM) [79], Artificial Neural Networks [157] (ANN) and Support Vector Regression (SVR) [46].

Random Forests are an ensemble-based learning method for classification and regression, and are made up of many decision trees. LSTMs are recurrent Neural Networks which remember values over arbitrary time intervals. Multilayer Perceptrons are a popular type of neural network which consist of a minimum of three layers and can be used to make non-linear predictions. SVRs are supervised learning models which analyse data for regression analysis.

To improve forecasting results, *k*-means clustering of smart meter data was evaluated. An average 24-hour electricity load profile per customer was calculated, and the result used for clustering. The clustered sub-system is then aggregated and separate models trained on these aggregates. The yearly, weekly and daily periodicity of electricity load is accounted for by feature vectors. Once forecasts for each cluster are made using the individual models, the results are aggregated for the final predictions. These predictions are compared to the actual results and the accuracy measured using mean absolute percentage error (MAPE) and mean absolute scaled error (MASE).

This paper provides researchers and utilities with methods to maximise forecasting accuracy through the selection of machine learning and clustering algorithms.

This paper is structured as follows. In Section 2 we explore related work of load forecasting. The experiments and their evaluation are discussed in Section 3. The results are discussed in Section 4. In Section 5 we conclude and consider future directions for this work.

5.10 Related Work

The forecasting of aggregated and clustered electricity demand has been the focus of a considerable amount of research in recent years. The research can generally be classified into two classes, Artificial Intelligence (AI) techniques [105, 122, 135] and classical time series approaches [**?** 86, 125]. For the purposes of our paper we have reviewed artificial intelligence techniques. Please refer to appendix **??** to explore the literature related to classical time series approaches.

Singh *et al.* produced a review of load forecasting techniques and methodologies and reported that hybrid methods, which combine two or more different techniques, are gaining traction, as well as soft computing approaches (AI) such as genetic algorithms [145].

5.10.1 Artificial Intelligence Techniques

Dillon *et al.* presented a Neural Network for short-term load forecasting. Their Neural Network consisted of three-layers and used adaptive learning for training [45]. They proposed the use of weather information to augment their electricity load data. They found better results with the Adaptive Neural Network than with a linear model, or Non-Adaptive Neural Network. In contrast to Dillon our paper focuses on a Non-Adaptive Neural Network and does not take into account weather information.

Chen *et al.* used an Artificial Neural Network to predict electricity demand of three substations in Taiwan. They integrated temperature data into the model, and showed a higher degree of accuracy when forecasting demand in residential and commercial substations as opposed to industrial. This was due to the ability to model the high usage of air-conditioners in residential and commercial substations using temperature data [28]. In contrast to the work by Chen *et al.*, we focus on client-side prediction using smart meter data. We were, therefore, able to cluster the data based on load profile, as opposed to grouping based on geographical location.

5.10.2 Clustering

Multiple techniques have been proposed for the clustering of electricity load data prior to forecasting. Shu *et al.* and Nagi *et al.* propose a hybrid approach in which self-organizing maps are used to cluster the data, and Support Vector Regression is used for prediction [144, 122]. This technique proved robust for different data types. Shu showed that this hybrid approach out-performed a single SVR technique, whilst Nagi showed superior results to a traditional ANN system. In contrast to both Nagi *et al.* and Shu *et al.* our paper utilises *k*-means as the clustering algorithm.

Wijaya *et al.* demonstrated that implementing a certain number of clusters improved load-forecasting accuracy [163]. However, a study by Ilić *et al.*, showed that increasing the number of clusters did not improve accuracy [91].

5.11 Methodology

The work in this paper was run on a MacBook Pro with a quad-core 3.1GHz Intel Core i7 processor with 16 GB 1867 MHz DDR3 of RAM and a 500GB solid state drive (SSD).

5.11.1 Data Collection

Smart meter data obtained from the Irish Social Science Data Archive (ISSDA) was used in this study [52]. The Commission for Energy Regulation released a public dataset of anonymised smart meter data from the "*Electricity Smart Metering Customer Behaviour Trials*" [3]. This dataset is made up of over 5000 Irish homes and businesses and is sampled at 30-minute intervals.

The data was recorded between the 14th July 2009 and 31st December 2010. For the purposes of cross-validation this data was split into a training, validation, and testing set. The training set consisted of the first 9 months of data and used to train the models, the validation set consisted of the following 2 months of data and used to tune the hyperparameters, and the test set included the remaining 6 months and used for measuring error. These splits were chosen to balance the training data with the test data and give the models a chance to learn the periodicity inherent in a one year period. Due to the long training times for these algorithms, we worked with a sample of 709 individual Irish homes. However, due to the infrequent requirement to train these models, we believe our technique would be suited for the real life application.

Figure 5.10 displays four residential customer daily load profiles. It can be seen that whilst Customer 1 and Customer 2 have similar load profiles, Customer 3 and Customer 4 have significantly different load profiles. This demonstrates that, whilst electricity consumption changes per-person, it is possible to cluster similar customers by their load profiles.

5.11.2 Clustering

We propose that clustering similar customer load profiles and aggregating each cluster's electricity consumption improves the accuracy of the models. This is due to the fact that aggregated clustered loads decrease the stochasticity of the demand of the system, and therefore increase the predictive ability of the models.

The Euclidean distance and wavelet metrics were evaluated using hierarchical clustering [119]. However, k-means demonstrated to be the most robust and best-performing algorithm, and thus was chosen for use in this paper [**?**].

To cluster the data, each user's nine month electricity consumption in the training set was reduced to a single averaged 24 hour load profile (48 data points per customer). This was achieved by taking the average electricity consumed for each half hourly point of the day (eg. taking the mean of every 12-12:30pm point in the training set). We did not take into account the difference between weekend and weekdays for clustering. The data was then scaled so that different sized households, but with similar usage profiles were clustered together.

To select the optimum number of clusters (k) the test set was used for cross-validation. This allowed us to compare the results of each of the models and select k with the highest accuracy exhibited by mean absolute percentage error (MAPE) and mean absolute scaled error (MASE). In this paper k was varied between 1 and 7, which was chosen due to the fact that the error did not vary greatly past seven clusters. Multiple models were fit per cluster and used to predict the testing data.

To overcome local minima the k-means algorithm was run 1000 times and the most accurate partition chosen [96].

5.11.3 Aggregating Demand

Once each customer is assigned to their respective cluster, the total electricity consumed by each cluster is calculated. This is achieved by summing the electricity consumed at every time interval per cluster. This creates a partial system load. A model is trained on each of the different partial system loads, and the resultant forecasts are aggregated to generate the total load forecast. This forecast is then evaluated by calculating the MAPE and MASE for each model.

5.11.4 Feature Selection

Calendar Attributes

Due to the daily, weekly and annual periodicity of the electricity consumption, daily calendar attributes were deemed important to accurately model the problem. The calendar attributes included are: hour of the day, day of the month, day of the week, month, and public holidays.

Public holidays were used as features for the model due to the change in electricity consumption exhibited on these days.

We evaluate the increase in performance due to the modelling of calendar attributes in the results section.

Time Series Data

The time-series element was modelled using lagged data inputs. This was achieved using the previous 3 hours, the equivalent 3 hours from the previous day, and the equivalent 3 hours from the previous week.

Long Short-Term Memory neural networks remember values over arbitrary time intervals. And thus can remember short-term memory over a long period of time, for this reason, 5 lagged inputs of the previous two and a half hours were used as features to the Long Short-Term Memory network.

Data Representation

Numerical representations were used for the lagged data input. A single binary was used for public holidays. One hot encoding is a method which allows categorical variables to be distinguished from ordinal data. One hot encoding was used to encode day of the week and month of the year. Table **??** displays the input data for SVR, Neural Network and Random Forest.

5.11.5 Experiments

This section explores the experiments made to design the models. Cross-validation was used on the validation set of each of the models to tune the hyperparameters. Each of the models were then created five times per cluster to explore the variance of the results.

Support Vector Regression

The validation set was used to tune the hyperparameters and select the kernel of the SVR model.

The kernels compared were polynomial, radial basis function (RBF) and the linear kernel [24, 153]. They were chosen due to their popularity, support and speed of computation.

The parameter values are shown in Table **??**. The linear kernel produced the best results, and therefore chosen as the final model.

Input	Variable	Detail description
1	Hour	Single numeric input representing hour of the day
2	Day of month	Single numeric input representing day of the month
3-9	Day of week	Seven binary digits representing calendar information regarding day of the week
10-21	Month	Twelve binary digits representing calendar information regarding month
22-42	Lagged inputs	Twenty one numeric inputs representing lagged inputs of previous 3 hours, previous 3 hours of previous day including hour to be predicted, and previous 3 hours of previous week including hour to be predicted
43	Holiday	Single binary digit representing whether the day was a public holiday

Table 5.3 List of Input Data for Models

Kernel Type	Kernel Parameters	RMSE
Linear	No values	0.02102779
RBF	C=2, $\gamma = 0.016$	0.02444950
Polynomial	C=2, $d = 2, r = 2$	0.03145719

Table 5.4 Prediction Accuracy Based on Type of Kernel

Random Forest

The number of variables sampled as candidates at each split was tuned using the validation set.

The optimum number for this was 23. It is proposed that the value 23 was found to be optimum due to the 21 lagged inputs, which are crucial to learn the underlying nature of the time series.

Artificial Neural Network

The first step when creating an Artificial Neural Network is to design the architecture. In our case, the number of input neurons is set to 43 (see Table **??**). Only one output neuron is required, due to the fact that we are only forecasting one step (30 minutes) ahead.

To design the number of hidden layers the Levenberg-Marquardt technique was used. An optimal architecture with three hidden layers was obtained. The first layer contained two neurons, the second contained five, and the third contained four.

LSTM

The Levenberg-Marquardt techniques was once again used to select number of layers and number of memory units. Using this technique, the optimum number of layers was found to be 2 with 50 memory units each.

5.12 Results

To determine the optimal number of clusters a range of values for k were explored, thus, k was varied between one and seven. 28 forecasting models were therefore constructed per type of model. The models were fit five times to explore the variation in the output. The model accuracy was evaluated using both MAPE and MASE.

The results are shown in figure 5.11. These show that clustering similar users improves accuracy. The optimum value for k for every model was shown to be four. After this, the accuracy diminishes slightly. The error bars shown in Figure 5.11 display a slight variance in MAPE for SVRs, ANNs and Random Forests. However, the MAPE of the LSTMs seem to vary by up to 11%.

The MASE metric also demonstrated that four clusters were optimal for SVR, Neural Networks and Random Forests.

The impact of using calendar attributes improves prediction accuracy by 6% for neural networks, 4% for Random Forests and 1% for SVR. For these results please see figure **??** in appendix **??**.

It is proposed that the optimum value for k cluster centres was four due to the distinct patterns observed in each of the clusters. At $k = 5$ one of the distinct clusters is split, and leads to an increase in stochasticity. At $k = 3$ the stochasticity is also increased by the aggregation of load profiles which are dissimilar.

However, the optimum number of clusters will vary for different datasets. Differing geographies may display varying usage characteristics due to culture, weather or social norms.

The results demonstrate that SVR, Random Forests and the ANN have similar accuracy, and adequately predict electricity consumption. The LSTM shows a similar pattern in increasing accuracy with number of clusters, but performs worse than the other models. The Random Forest seems to outperform each of the other models. This may be due to the internal operation of the Random Forest which undertakes its own cross-validation using out-of-bag samples.

Figure 5.12 displays actual electricity consumption versus predicted results. It shows that the LSTM model predicts a similar value in the next time step as the previous time step, which would explain its inferior results to the other models.

The training times were tested by timing how long the models took to fit for four clusters. The Support Vector Regression took less time than all of the other methods, whereas the LSTM took the longest. The Support Vector Regression model required 3 minutes and 18 seconds to run. The Random Forest required 14 minutes and 58 seconds. The Artificial Neural Network required 17 minutes and 48 seconds, whilst the LSTM took 21 minutes 11 seconds to run.

The time to make a single prediction was recorded at sub microseconds and therefore deemed negligible for our use-case.

5.13 Conclusion

The availability of data produced by smart meters enables network operators to gain greater insights into their customer behaviour and electricity usage. We demonstrated that implementing the k-means clustering algorithm to group similar customers improved the accuracy of the models tested. Distinct models were trained for each cluster and the individual forecasts aggregated for the total aggregated forecast. It was found that Random Forests outperformed all other models at each level of clustering. The optimum number of clusters was found to be four. Whilst the dataset used focused on residential data it is expected that applying a similar clustering technique on commercial properties would have a similar effect.

It is considered that monthly retraining of the models would be required to ensure continued accuracy. However, this is not expected to be a problem due to the short time time required for model training. Once trained, the prediction times are negligible.

In future work, we will look into the features that best aid in the forecasting of electricity consumption, try a wider variety of models in an ensemble manner and try different clustering techniques such as self-organizing maps (SOM) to obtain better accuracy results. We will also compare a greater variety of forecasting metrics.

5.14 The impact of online machine-learning methods on long-term investment decisions and generator utilization in electricity markets

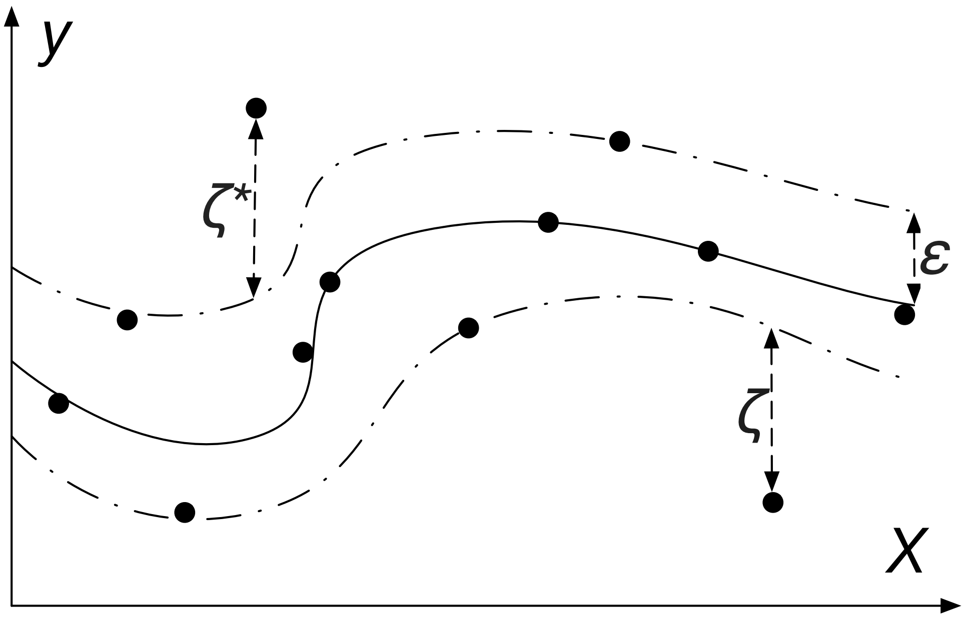


Fig. 5.2 ε-insensitive band for SVR [144].

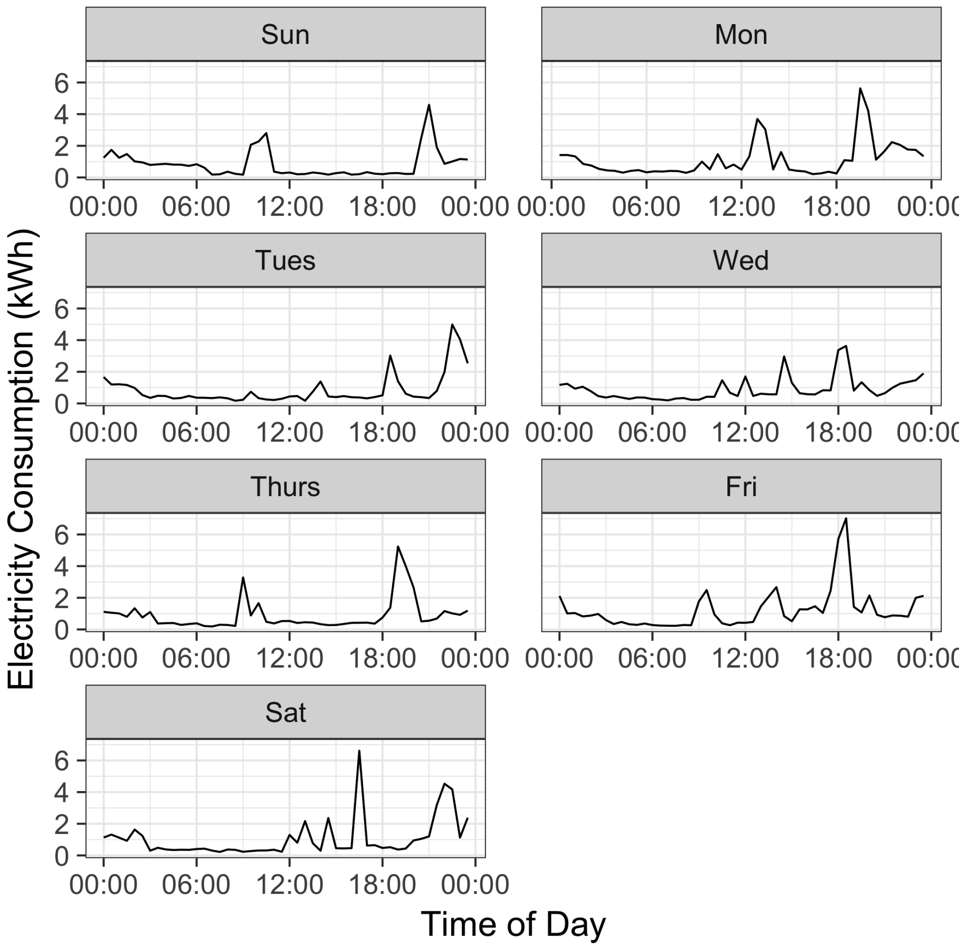


Fig. 5.3 Daily load profiles of a single customer over a week between 20th July 2009 and 27th July 2009.

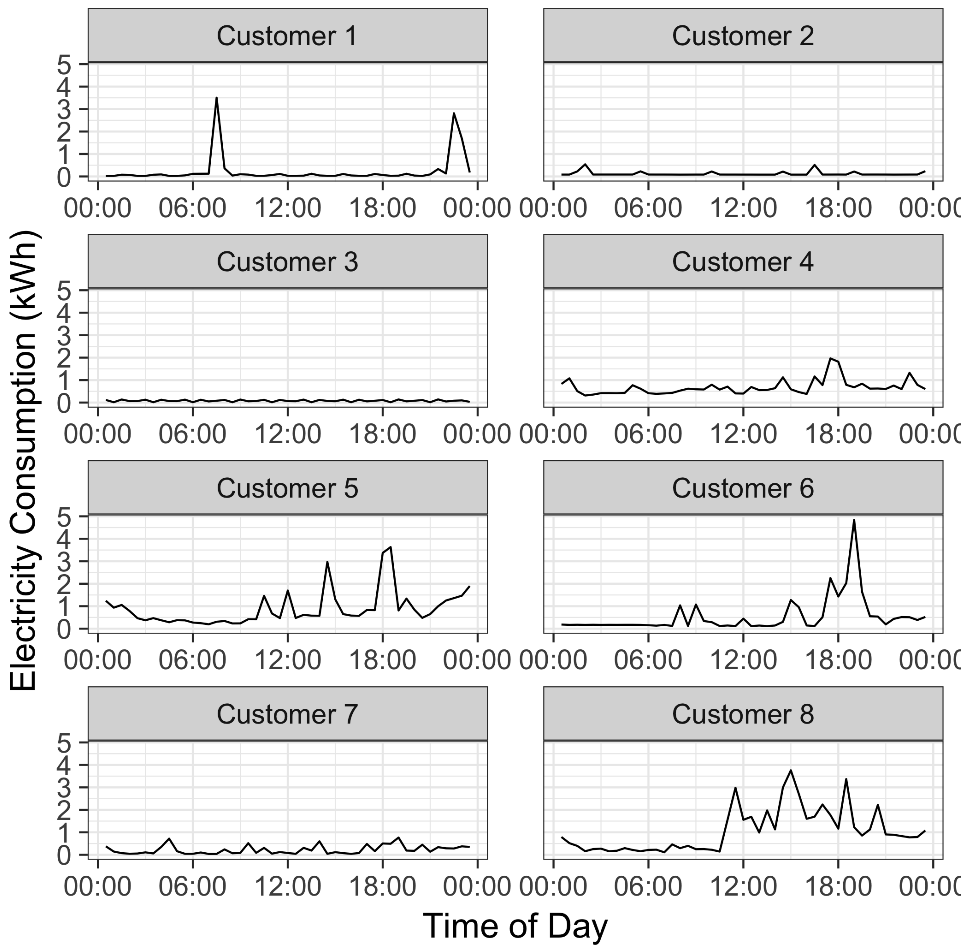


Fig. 5.4 Daily load profiles of different customers over a single day on the 22nd June 2009.

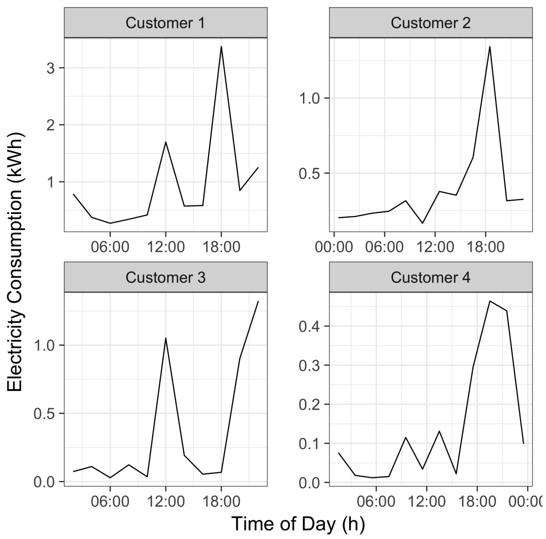


Fig. 5.5 Figure showing similar load profiles for four different customers on the 22nd July 2009.

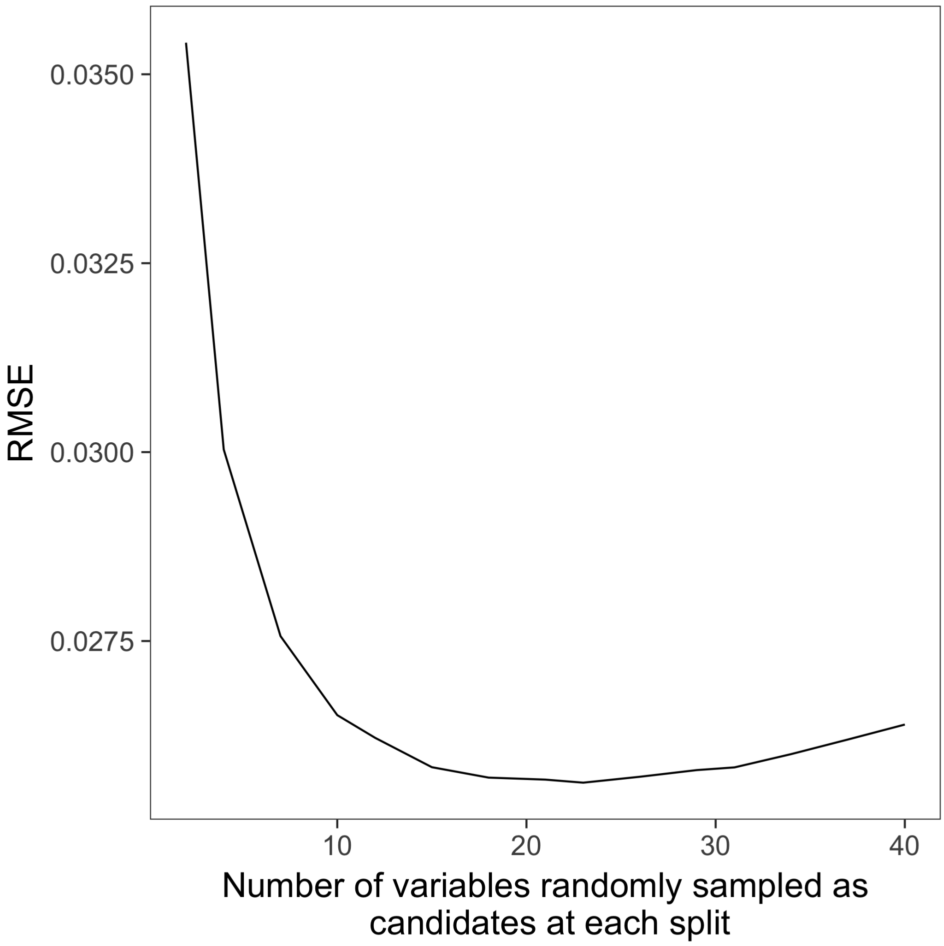


Fig. 5.6 RMSE vs Number of variables randomly sampled as candidates at each split in the Random Forest model.

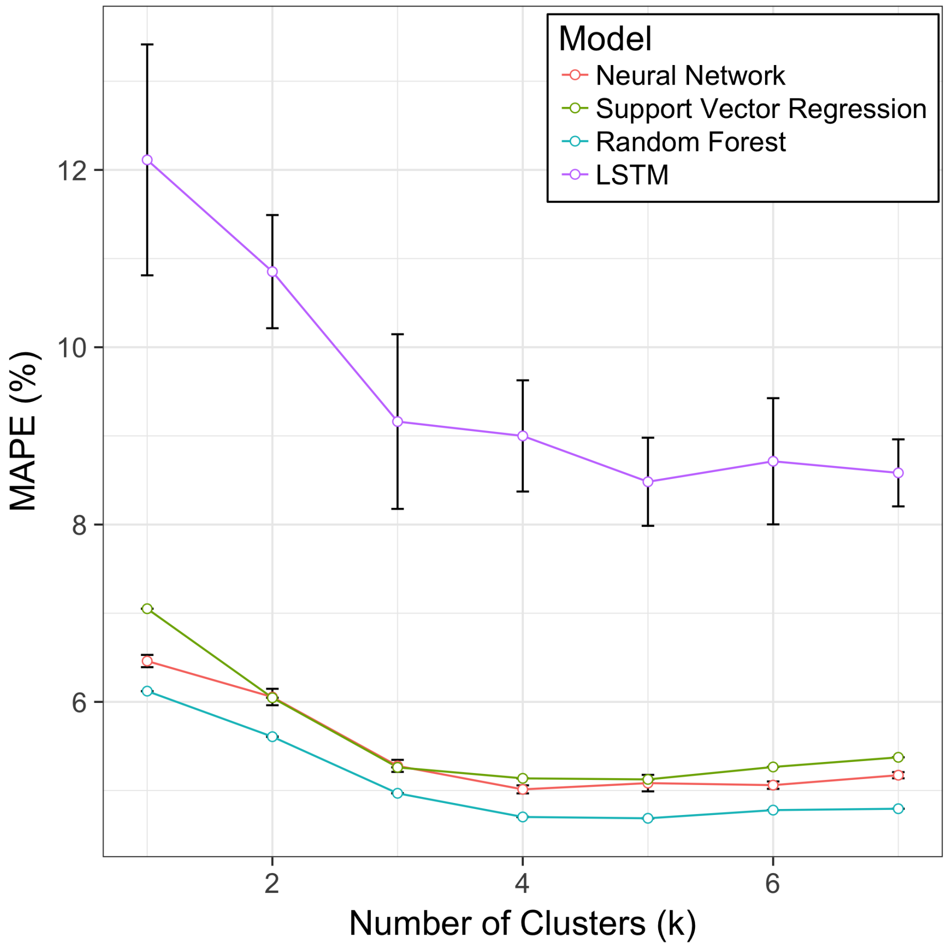


Fig. 5.7 Comparison of accuracy of models forecasting electricity with varying number of clusters.

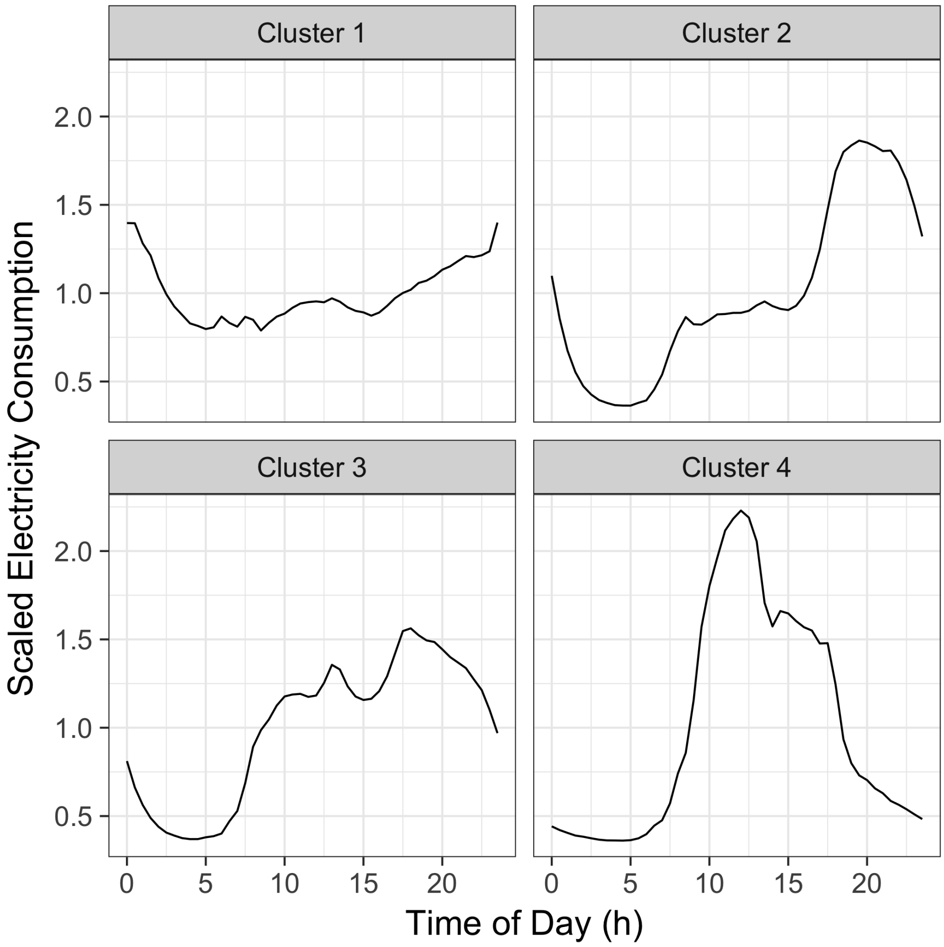


Fig. 5.8 Average load profile for each cluster.

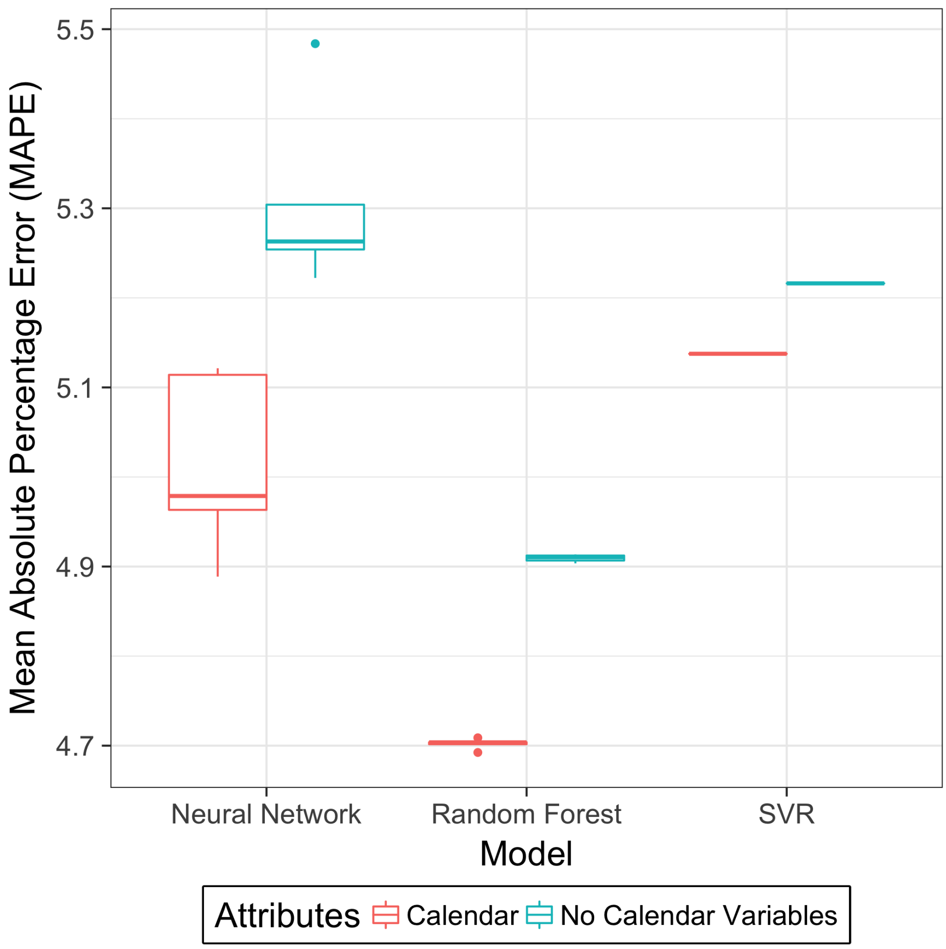


Fig. 5.9 Comparison of accuracy of models with or without calendar attributes.

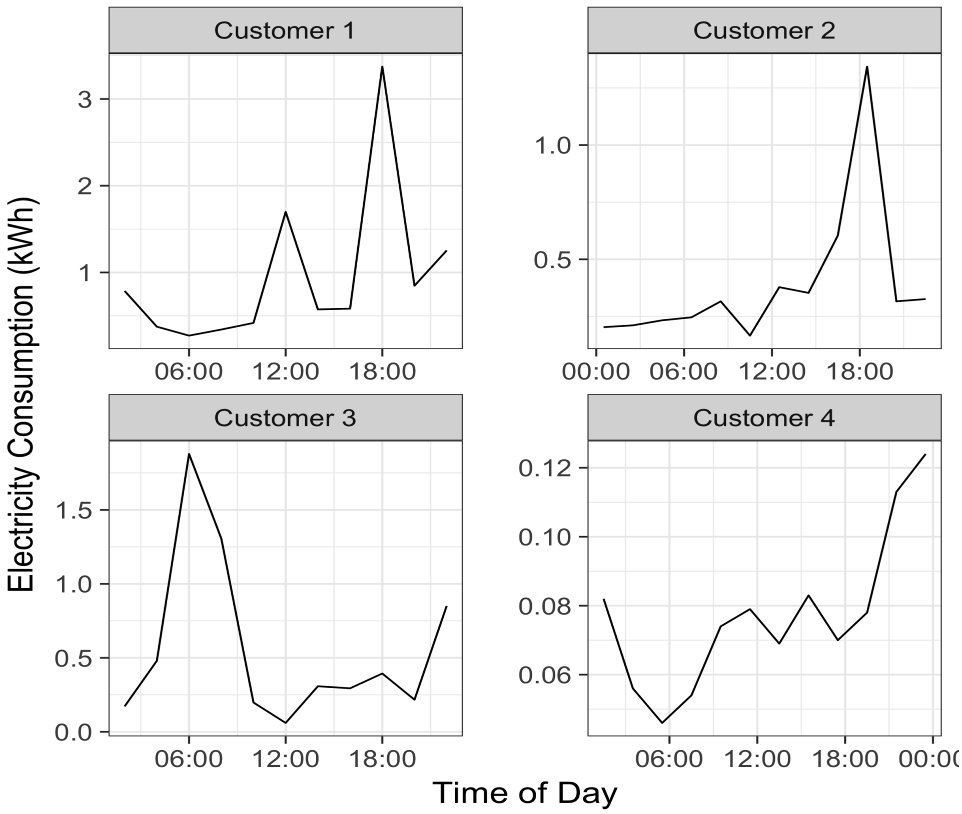


Fig. 5.10 Figure showing daily load profiles for four different customers on the 22nd July 2009.

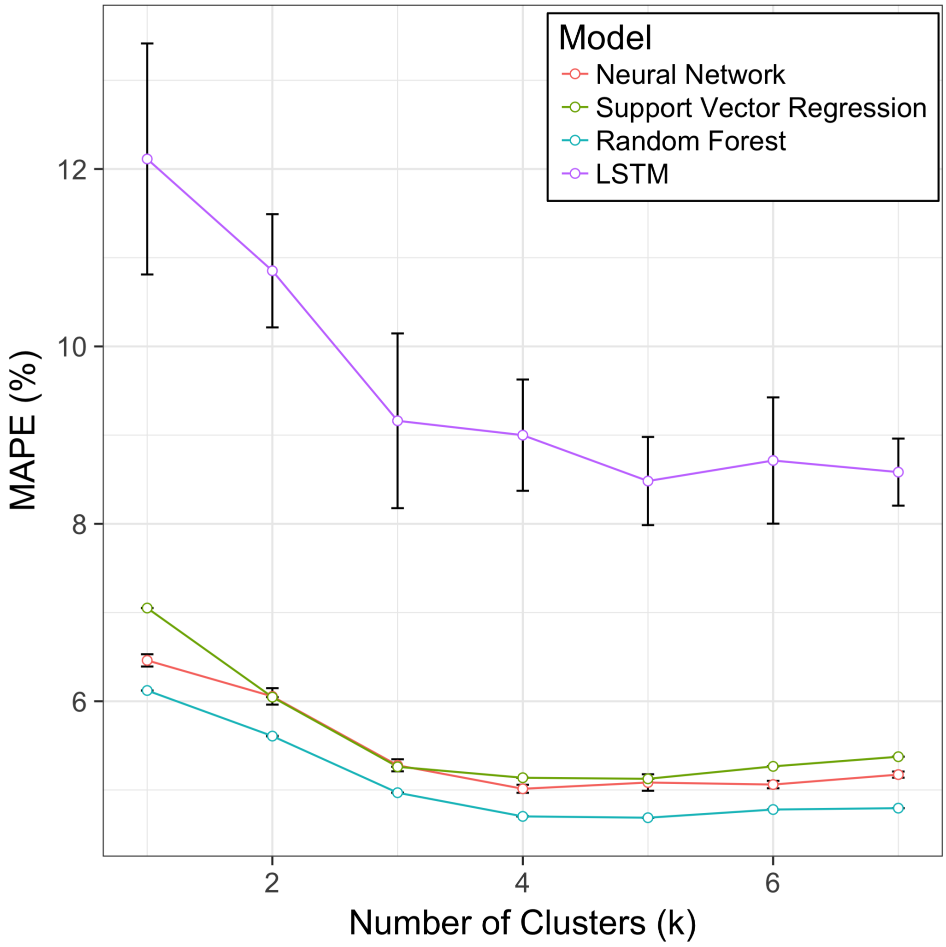


Fig. 5.11 Comparison of accuracy of models forecasting electricity with varying number of clusters.

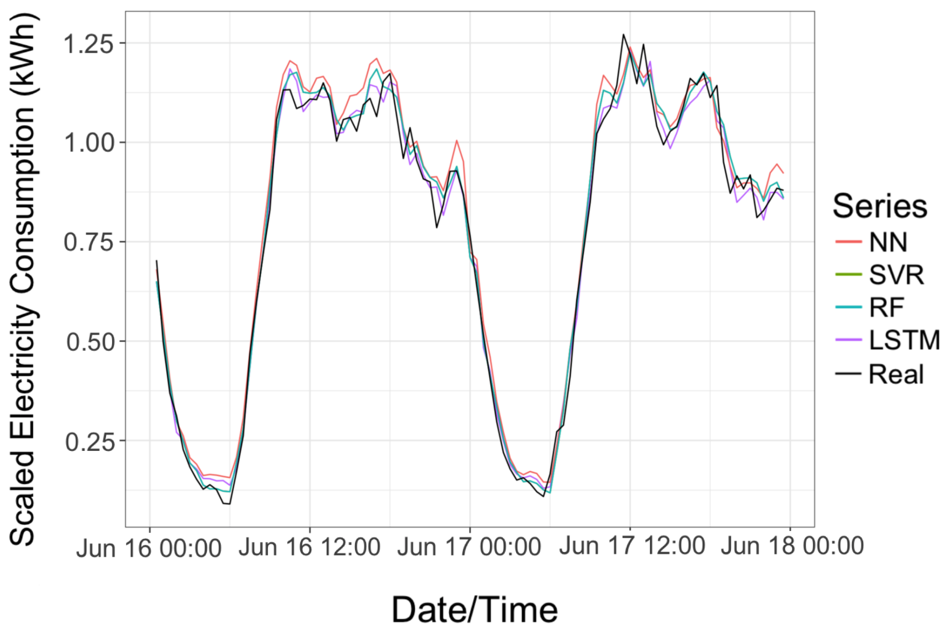


Fig. 5.12 Real electricity consumption versus predicted electricity consumption between June 16th and June 18th 2010.

Abstract

Electricity supply must be matched with demand at all times. This helps reduce the chances of problems with load frequency control and the chances of electricity blackouts. To gain a better understanding of the load that is likely to be required over the next 24 hours, estimations under uncertainty are needed. This is especially difficult in a decentralized electricity market with many micro-producers who are not under central control.

In this paper, we investigate the impact of 11 offline learning and five online learning algorithms to predict the electricity demand profile over the next 24 hours. We achieve this through integration within the long-term agent-based model, ElecSim. These algorithms include multilayer perceptron neural networks, support vector regression and linear regression. By predicting the electricity demand profile over the next 24 hours, we can simulate the predictions made for a day-ahead market. Once we have made these predictions, we sample from the residual distributions and perturb the electricity market demand using the simulation, ElecSim. This enables us to understand the impact of errors on the long-term dynamics of a decentralized electricity market.

Our results show that we are able to reduce the mean absolute error by 30% using an online algorithm when compared to the best offline model, whilst reducing the required tendered national grid reserve required. Such a reduction allows for a smaller required tendered grid reserve, saving costs and emissions. We also show that large errors in prediction accuracy have a disproportionate error on investments made over a 17-year time frame, as well as electricity mix.

5.15 Introduction

The integration of higher proportions of intermittent renewable energy sources (IRES) in the electricity grid will mean that the forecasting of electricity demand will become increasingly important and challenging. Examples of IRES are solar panels and wind turbines, which fluctuate in terms of power output based on localized wind speed and solar irradiance. However, as supply must meet demand at all times and the fact that IRES are less predictable than dispatchable energy sources such as coal and combined-cycle gas turbines (CCGTs) this means extra attention must be made in predicting future demand if we wish to keep, or better reduce, the current frequency of blackouts [114]. A dispatchable source is one that can be turned on and off by human control and therefore, able to adjust output just in time, at a moment convenient for the grid.

Typically, peaker plants, such as reciprocal gas engines, are used to fill fluctuations in demand, that had not been previously planned for. Specifically, peaker plants meet the peaks in demand where other cheaper options are at full capacity. These peaker plants are typically expensive to run and have higher greenhouse gas emissions than their non-peaker counterparts [118]. Whilst peaker plants are also dispatchable plants, not all dispatchable plants are peaker plants. For example coal, which is a dispatchable plant, is run as a base load plant, due to its inability to deal with the fluctuating conditions required of a peaker plant.

To reduce reliance on peaker plants, it is helpful to know how much electricity demand there will be in the future so that more efficient plants can be used to meet this expected demand. This is so that these more efficient plants can be brought up to speed at a time suitable to match the demand. Forecasting a day into the future is especially useful in decentralized electricity markets which have day-ahead markets. Decentralized electricity markets are ones where electricity is provided by multiple generation companies, as opposed to a centralized source, such as a government. To aid in this prediction, machine learning and statistical techniques have been used to accurately predict demand based on several different factors and data sources [102], such as weather [84], day of the week [8] and holidays [159].

Various studies have looked at predicting electricity demand at various horizons [145, 86, 10]. However, the impact of poor demand predictions on the long-term electricity mix has been studied to a lesser degree.

In this paper, we compare several machine learning and statistical techniques to predict the energy demand for each hour over the next 24-hour horizon. We chose to predict over the next 24 hours to simulate a day-ahead market, which is often seen in decentralized electricity markets. However, our approach could be utilized for differing time horizons. In addition to this, we use our long-term agent-based model, ElecSim [101, 104], to simulate the impact of different forecasting methods on long-term investments, power plant usage and carbon emissions for the years 2018 through 2035 in the United Kingdom. Our approach, however, is generalizable to any country through parametrization of the ElecSim model.

As part of our work, we utilize online learning methods to improve the accuracy of our predictions. Online learning methods can learn from novel data while maintaining what was learnt from previous data. Online learning is useful for non-stationary datasets, and time-series data where recalculation of a model would take a prohibitive amount of time. Offline learning methods, however, must be retrained every time new data is added. Online approaches are constantly updated and do not require significant pauses while the offline training is being re-run.

We trial different algorithms and train different models for different times of the year. Specifically, we train different models for the different seasons. We also split weekdays and train both weekends and holidays together. This is due to the fact that holidays and weekend exhibit similar load profiles due to the reduction in industry electricity use and an increase in domestic. This enables a model to become good at a specific subset of the data which share similar patterns, as opposed to having to generalize to all of the data. Examples of the algorithms used are linear regression, lasso regression, random forests, support vector regression, multilayer perceptron neural network, box-cox transformation linear regression and the passive aggressive model.

We expect a-priori that online algorithms will outperform the offline approach. This is due to the fact that the demand time-series is non-stationary, and thus changes sufficiently over time. In terms of the models, we presume that the machine learning algorithms, such as neural networks, support vector regression and random forests will outperform the statistical methods such as linear regression, lasso regression and box-cox transformation regression. We expect this due to the fact that machine learning has been shown to be able to learn more complex feature representations than statistical methods [145]. In addition, our previous work has shown that the random forest was able to outperform neural networks, support vector regression and long short term memory neural networks (LSTM) [103].

However, it should be noted, that such a-priori intuiton, is no substitute for analytical evidence and can (and has) been shown to be wrong in the past, due to imperfect knowledge of the data and understanding of some of the black box models, such as neural networks.

Using online and offline methods, we take the error distributions, or residuals, and fit a variety of distributions to these residuals. We choose the distribution with the lowest sum of squared estimate of errors (SSE). SSE was chosen as the metric to ensure that both positive and negative errors were treated equally, as well as ensuring that large errors were penalized more than smaller errors. We fit over 80 different distributions, which include the Johnson Bounded distribution, the uniform distribution and the gamma distribution. The distribution that best fits the respective residuals is then used and sampled from to adjust the demand in the ElecSim model. We then observe the differences in carbon emissions, and which types of power plants were both invested in and utilized, with each of the different statistical and machine learning methods. To the best of our knowledge, this is the most comprehensive evaluation of online learning techniques to the application of day-ahead load forecasting as well as assessing the impacts of the errors that these models produce on the long-term electricity market dynamics.

We show that online learning has a significant impact on reducing the error for predicting electricity consumption a day ahead when compared to traditional offline learning techniques, such as multilayer perceptron artificial neural networks, linear regression, extra trees regression and support vector regression, which are models used in the literature [114, 6, 27]. For a full list of algorithms used in this work see Table 5.5.

We show that the forecasting algorithm has a non-negligible impact on carbon emissions and use of coal, onshore, photovoltaics, reciprocal gas engines and CCGT. Specifically, the amount of coal, photovoltaics, and reciprocal gas used from 2018 to 2035 was proportional to the median absolute error, while both onshore and offshore wind are inversely proportional to the median absolute error.

Total investments in coal, offshore and photovoltaics are proportional to the median absolute error, while investments in CCGT, onshore and reciprocal gas engines are inversely proportional.

The contributions of this work are:

1. The evaluation of different online and offline learning models to forecast the electricity demand profile 24 hours ahead.

2. Evaluation of poor predictive ability on the long-term electricity market in the UK through the perturbation of demand in the ElecSim simulation.

In Section 5.16, we review the literature, including other uses of online learning algorithms and an application to electricity markets. We introduce the dynamics of the ElecSim simulation as well as the methods used in Section 5.17. We demonstrate the methodology undertaken in Section 5.18. In Section 6.6 we demonstrate our results, followed by a discussion in Section 5.20. We conclude our work in Section 6.7.

5.16 Literature Review

Multiple papers have looked at demand-side forecasting [145]. These include both artificial intelligence [105, 122, 135] and statistical techniques [86, 125]. To the best of our knowledge, the impact of online learning has been discussed with less frequency. In addition to this, our research models the impact of the performance of different algorithms on investments made, electricity sources dispatched and carbon emissions over a 17 year period. To model this, we use the long-term electricity market agent-based model, ElecSim.

5.16.1 Offline learning

Multiple electricity demand forecasting studies have been undertaken for offline learning [27, 64, Ghofrani et al.]. Studies have been undertaken using both smart meter data, as well as with aggregated demand, similar to the work in this paper. Smart meters are a type of energy meter installed in each house, which monitor electricity usage at short intervals, such as every 15 or 30 minutes.

Fard *et al.* propose a new hybrid forecasting method based on the wavelet transform, autoregressive integrated moving average (ARIMA) and artificial neural network (ANN) [49]. The ARIMA model is utilized to capture the linear component of the time series, with the residuals containing the non-linear components. The non-linear parts are decomposed using the discrete wavelet transform, which finds the sub-frequencies. These residuals are then used to train an ANN to predict the future residuals. Finally, the ARIMA and ANNs outputs are summed. Their results show that this technique can improve the load forecasting results.

Humeau *et al.* compare MLPs, SVRs and linear regression at predicting smart meter data [87]. They aggregate different households and observe which models work the best at each aggregate level. They find that linear regression outperforms both MLP and SVR when forecasting individual households. However, after aggregating over 32 households, SVR outperforms linear regression.

Quilumba *et al.* also apply machine learning techniques to individual households' electricity consumption by aggregation [49]. To achieve this aggregation, they use *k*-means clustering to aggregate the households to improve their forecasting ability. The authors also use a neural network based model for forecasting, and show that the number of optimum clusters for forecasting is dependent on the data, with three clusters optimal for a particular dataset, and four for another.

In our previous work we evaluate the performance of ANNs, random forests, support vector regression and long short-term memory neural networks [102]. We utilize smart meter data, and cluster by household using the k-means clustering algorithm to aggregate groups of demand. We find that through this clustering we are able to reduce the error, with the random forest performing the best.

5.16.2 Online learning

There have been several studies in diverse applications on the use of online machine learning to predict time-series data, however, to the best of our knowledge there are limited examples where this is applied to electricity markets. In our work, we trial a different set of algorithms to our problem. Due to time constraints, we do not trial the additional techniques discussed in this literature review within our paper.

Johansson *et al.* apply online machine learning algorithms for heat demand forecasting [97]. They find that their demand predictions display robust behaviour within acceptable error margins. They find that artificial neural networks (ANNs) provide the best forecasting ability of the standard algorithms and can handle data outside of the training set. Johansson *et al.*, however, do not look at the long-term effects of different algorithms on their application.

Baram *et al.* combine an ensemble of active learners by developing an active-learning master algorithm [12]. To achieve this, they propose a simple maximum entropy criterion that provides effective estimates in realistic settings. Their active-learning master algorithm is empirically shown to, in some cases, outperform the best algorithm in the ensemble on a range of classification problems.

Schmitt *et al* also extends on existing algorithms through an extension of the FLORA algorithm in [142**?**]. The FLORA algorithm generates a rule-based model, which has the ability to make binary decisions. Their FLORA-MC enhances the FLORA algorithm for multi-classification and numerical input values. They use this algorithm for an ambient computing application. Ambient computing is where computing and communication merges into everyday life. They find that their model outperforms traditional offline learners by orders of magnitude.

Similarly to us, Pindoriya *et al.* trial several different machine learning methods such as adaptive wavelet neural network (AWNN). They find that AWNN has good prediction properties when compared to other forecasting techniques such as wavelet-ARIMA, multilayer perceptron (MLP) and radial basis function (RBF) neural networks as well as the fuzzy neural network (FNN).

Goncalves Da Silva *et al.* show the effect of prediction accuracy on local electricity markets [63]. To this end, they compare forecasting of groups of consumers in comparison to single individuals. They trial the use of the Seasonal-Naïve and Holt-Winters algorithms and look at the effect that the errors have on trading in an intra-day electricity market of consumers and prosumers. They found that with a photovoltaic penetration of 50%, over 10% of the total generation capacity was uncapitalized and roughly 10, 25 and 28% of the total traded volume were unnecessary buys, demand imbalances and unnecessary sells respectively. This represents energy that the participant has no control. Uncapitalized generation capacity is where a participant could have produced energy, however, it was not sold on the market. Additionally, due to forecast errors, the participant might have sold less than it should have. Our work, however, focuses on a national electricity market, as opposed to a local market.

5.17 Material

5.17.1 Machine Learning

Machine learning is a methodology for finding and describing structural patterns in data [164]. Offline learning models are trained with the data availalable at a single point in time. With non-stationary data where underlying distributions change, the model must be retrained at periodic intervals, determined by how quickly the model goes out of step with the true data. With online learning, the model is able to retrain every time a new data point becomes available, without having to retrain the entire model. This makes these models good for time-series data which exhibit moderate to significant non-stationary properties, such as electricity demand profiles.

5.17.2 Online learning

Examples of online learning algorithms are Passive Aggressive (PA) Regressor [37], Linear Regression, Box-Cox Regressor [19], K-Neighbors Regressor [53] and Multilayer perceptron regressor [78]. For our work, we trial the stated algorithms, in addition to a host of offline learning techniques. The offline techniques trialled were Lasso regression [154], ridge regression [**?**], Elastic Net [57], Least Angle Regression [47], Extra Trees Regressor [47], Random Forest Regressor [21], AdaBoost Regressor [56], Gradient Boosting Regressor [58] and Support vector regression [35]. We chose the boosting and random forest techniques due to previous successes of these algorithms when applied to electricity demand forecasting [103]. We trialled the additional algorithms due to availability of these algorithms using the scikit-learn package and online learning package, Creme [**? ?**].

5.17.3 Linear regression models

Linear regression is a linear approach to modelling the relationship between a dependent variable and one or more independent variables. Linear regressions can be used for both online and offline learning. In this work, we used them for both online and offline learning. Linear regression models are often fitted using the least squares approach. The least squares approach minimizes the sum of the squares of the residuals.

Other methods for fitting linear regressions are by minimizing a penalized version of the least squares cost function, such as in ridge and lasso regression [154**?**]. Ridge regression is a useful approach for mitigating the problem of multicollinearity in linear regression. Multicollinearity is where one predictor variable can be linearly predicted from the others with a high degree of accuracy. This phenomenon often occurs in models with a large number of parameters.

In ridge regression, the OLS loss function is augmented so that we not only minimize the sum of squared residuals but also penalized the size of parameter estimates, in order to shrink them towards zero:

$$L_{ridge}(\hat{\beta}) = \sum_{i=1}^{n}(y_i - x_i'\hat{\beta})^2 + \lambda \sum_{j=1}^{m} \hat{\beta}_j^2 = ||y - X\hat{\beta}||^2 + \lambda||\hat{\beta}||^2. \tag{5.17}$$

Where λ is the regularization penalty which can be chosen through cross-validation, or the value that minimizes the cross-validated sum of squared residuals, for instance. n is the number of observations of the response variable, Y, with a linear combination of m predictor variables, X, and we solve for $\hat{\beta}$, where $\hat{\beta}$ are the OLS parameter estimates.

Lasso is a linear regression technique which performs both variable selection and regularization. It is a type of regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, such as the mean. The lasso model encourages models with fewer parameters. This enables the selection of models with fewer numbers of parameters, or automate the process of variable selection.

Under Lasso the loss is defined as:

$$L_{lasso}(\hat{\beta}) = \sum_{i=1}^{n}(y_i - x_i'\hat{\beta})^2 + \lambda \sum_{j=1}^{m} |\hat{\beta}_j|. \tag{5.18}$$

The only difference between lasso and ridge regression is the penalty term.

Elastic net is a regularization regression that linearly combines the penalties of the lasso and ridge methods. Specifically, Elastic Net aims to minimize the following loss function:

$$L_{enet}(\hat{\beta}) = \frac{\sum_{i=1}^{n}(y_i - x_i'\hat{\beta})^2}{2n} + \lambda(\frac{1-\alpha}{2} \sum_{j=1}^{m} \hat{\beta}_j^2 + \alpha \sum_{j=1}^{m} |\hat{\beta}_j|), \tag{5.19}$$

where α is the mixing parameter between ridge ($\alpha = 0$) and lasso ($\alpha = 1$). The two parameters λ and α can be tuned.

Least Angle Regression (LARS) provides a mean of producing an estimate of which variables to include in a linear regression, as well as their coefficients.

5.17.4 Decision tree-based algorithms

The decision tree is a model which goes from observations to output using simple decision rules inferred from data features [**?**]. To build a regression tree, recursive binary splitting is used on the training data. Recursive binary splitting is a greedy top-down algorithm used to minimize the residual sum of squares. The RSS, in the case of a partitioned feature space with M partitions, is given by:

$$RSS = \sum_{m=1}^{M} \sum_{i \in R_m} (y - \hat{y}_{R_m})^2. \tag{5.20}$$

Where y is the value to be predicted, $\hat{y}$ is the predicted value for partition R_m.

Beginning at the top of the tree, a split is made into two branches. This split is carried out multiple times and the split is chosen that minimizes the current RSS. To obtain the best sequence of subtrees cost complexity, pruning is used as a function of α. α is a tuning parameter that balances the depth of the tree and the fit to the training data. This parameter can be tuned using cross-validation.

The AdaBoost training process selects only the features of a model known to improve the predictive power of the model [56]. By doing this, the dimensionality of the model is reduced and can improve compute time. This can be used in conjunction with multiple different models. In our paper, we utilized the decision tree based algorithm with AdaBoost.

Random Forests are an ensemble learning method for classification and regression [21]. Ensemble learning methods use multiple learning algorithms to obtain better predictive performance. They work by constructing multiple decision trees at training time, and outputting the predicted value that is the mode of the predictions of the individual trees.

To ensure that the individual decision trees within a Random Forest are not correlated, bagging is used to sample from the data. Bagging is the process of randomly sampling with replacement of the training set and fitting the trees. This has the benefit of reducing the variance of the model without increasing the bias.

Random Forests differ in one way from this bagging procedure. Namely, using a modified tree learning algorithm that selects, at each candidate split in the learning process, a random subset of the features, known as feature bagging. Feature bagging is undertaken due to the fact that some predictors with a high predictive ability may be selected many times by the individual trees, leading to a highly correlated Random Forest.

ExtraTrees adds one further step of randomization [47]. ExtraTrees stands for extremely randomized trees. There are two main differences between ExtraTrees and Random Forests. Namely, each tree is trained using the whole learning sample (And not a bootstrap sample), and the top-down splitting in the tree learner is randomized. That is, instead of computing an optimal cut-point for each feature, a random cut-point is selected from a uniform distribution. The split that yields the highest score is then chosen to split the node.

5.17.5 Gradient Boosting

Gradient boosting is also an ensemble model [58]. Gradient boosting optimizes a cost-function over function space by iteratively choosing a function that points in the negative gradient descent direction, known as a gradient descent method.

5.17.6 Support vector regression

Support vector regression is an algorithm which finds a hyperplane and decision boundary to map an input domain to an output [35]. The hyperplane is chosen by minimizing the error within a certain tolerance.

Suppose we have the training set: $(x_1,y_1),\dots,(x_i,y_i),\dots,(x_n,y_n)$, where x_i is the input, and y_i is the output value of x_i. Support Vector Regression solves an optimization problem [144, 27], under given parameters $C > 0$ and $\varepsilon > 0$, the form of support vector regression is [46]:

$$\min_{\omega,b,\xi,\xi^*} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{5.21}$$

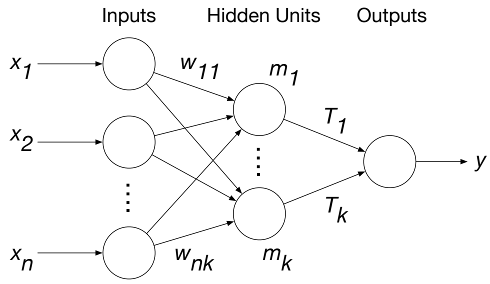


Fig. 5.13 A three-layer feed forward neural network.

subject to

$$\begin{aligned} y_i - (\omega^T \phi(x_i) + b) &\leq \varepsilon + \xi_i^*, \\ (\omega^T \phi(x_i) + b) - y_i &\leq \varepsilon + \xi_i, \\ \xi_i, \xi_i^* &\geq 0, i = 1, \ldots, n \end{aligned} \tag{5.22}$$

x_i is mapped to a higher dimensional space using the function ϕ. The ε-insensitive tube $(\omega^T \phi(x_i) + b) - y_i \leq \varepsilon$ is a range in which errors are permitted. ξ_i and ξ_i^* are slack variables which allow errors for data points which fall outside of ε. This enables the optimization to take into account the fact that data does not always fall within the ε range [147].

The constant $C > 0$ determines the trade-off between the flatness of the support vector function. ω is the model fit by the SVR. The parameters which control regression quality are the cost of error C, the width of the tube ε, and the mapping function ϕ [144, 27].

5.17.7 K-Neighbors Regressor

K-Neighbors regression is a non-parametric method used for regression [53]. The input consists of a new data point, and the algorithm finds the k closest training examples in the feature space. The output is the average value of the k nearest neighbours.

5.17.8 Multilayer perceptron

A neural network can be used in both offline and online cases. In this work, we used them for both online and offline.

Artificial Neural Networks are a model which can model non-linear relationships between input and output data [7]. A popular neural network is a feed-forward multilayer perceptron. Fig. 5.13 shows a three-layer feed-forward neural network with a single output unit, k hidden units, n input units. w_{ij} is the connection weight from the i^{th} input unit to the j^{th} hidden unit, and T_j is the connecting weight from the j^{th} hidden unit to the output unit [128]. These weights transform the input variables in the first layer to the output variable in the final layer using the training data.

For a univariate time series forecasting problem, suppose we have N observations $y_1, y_2, \ldots, y_N$ in the training set, and m observations in the test set, $y_{N+1}, y_{N+2}, \ldots, y_{N+m}$. In the test set and we are required to predict m periods ahead [128].

The training patterns are as follows:

$$y_{p+m} = f_W(y_p, y_{p-1}, \dots, y_1) \tag{5.23}$$
$$y_{p+m+1} = f_W(y_{p+1}, y_p, \dots, y_2) \tag{5.24}$$
$$\vdots$$
$$y_N = f_W(y_{N-m}, y_{N-m-1}, \dots, y_{N-m-p+1}) \tag{5.25}$$

where $f_W(\cdot)$ represents the MLP network and W are the weights. For brevity we omit W. The training patterns use previous time-series points, for example, $y_p, y_{p-1}, \dots, y_1$ as the time series is univariate. That is, we only have the time series in which we can draw inferences from. In addition, these time series points are correlated, and therefore provide information that can be used to predict the next time point.

The m testing patterns are

$$y_{N+1} = f_W(y_{N+1-m}, y_{N-m}, \dots, y_{N-m-p+2}) \tag{5.26}$$
$$y_{N+2} = f_W(y_{N+2-m}, y_{N-m+1}, \dots, y_{N-m-p+3}) \tag{5.27}$$
$$\vdots$$
$$y_{N+m} = f_W(y_N, y_{N-1}, \dots, y_{N-p+1}). \tag{5.28}$$

The training objective is to minimize the overall predictive mean sum of squared estimate of errors (SSE) by adjusting the connection weights. For this network structure the SSE can be written as:

$$SSE = \sum_{i=p+m}^{N} (y_i - \hat{y}_i) \tag{5.29}$$

where $\hat{y}_i$ is the prediction from the network. The number of input nodes corresponds to the number of lagged observations. Having too few or too many input nodes can affect the predictive ability of the neural network [128].

It is also possible to vary the hyperparameter, the number of input units. Typically, various different configurations of units are trialled, with the best configuration being used in production. The weights W in f_W are trained using a process called backpropagation, which uses labelled data and gradient descent to update and optimize the weights.

5.17.9 Online Algorithms

In this Section we discuss the algorithms which were used exclusively for online learning in this work.

5.17.10 Box-Cox regressor

In this subsection, we discuss the Box-Cox regressor. Ordinary least square is a method for estimating the unknown parameters in a linear regression model. It estimates these unknown parameters by the principle of least squares. Specifically, it minimizes the sum of the squares of the differences between the observed variables and those predicted by the linear function.

The ordinary least squares regression assumes a normal distribution of residuals. However, when this is not the case, the Box-Cox Regression may be useful [19]. It transforms the dependent variable using the Box-Cox Transformation function and employs maximum likelihood estimation to determine the optimal level of the power parameter lambda. The Box-Cox Regression requires that no dependent variable has any negative values.

5.17.11 Passive-Aggressive regressor

The goal of the Passive-Aggressive (PA) algorithm is to change itself as little as possible to correct for any mistakes and low-confidence predictions it encounters [37]. Specifically, with each example PA solves the following optimisation [116]:

$$w_{t+1} \leftarrow argmin \frac{1}{2} ||w_t - w||^2 \tag{5.30}$$

$$s.t. \;\; y_i(w \cdot x_t) \geq 1. \tag{5.31}$$

Where x_t is the input data and y_i the output data, and w_t are the weights for the PA algorithm. Updates occur when the inner product does not exceed a fixed confidence margin - i.e., $y_i(w \cdot x_t) \geq 1$. The closed-form update for all examples is as follows:

$$w_{t+1} \leftarrow w_t + \alpha_t y_t x_t \tag{5.32}$$

where

$$\alpha_t = max \left\{ \frac{1 - y_t(w_t \cdot x_t)}{||x_t||^2}, 0 \right\}. \tag{5.33}$$

a_t is derived from a derivation process which uses the Lagrange multiplier. For full details of the derivation see [37].

5.17.12 Long-term Energy Market Model

In order to test the impact of the different residual distributions, we used the ElecSim simulation developed by Kell *et al.*, ElecSim [101, 104]. ElecSim is an agent-based model which mimics the behaviour of decentralized electricity markets. In this paper, we parametrized the model with data of the United Kingdom in 2018. This enabled us to create a digital twin of the UK electricity market and project forward. The data used for this parametrization included power plants in operation in 2018 and the funds available to the generation companies [51, 40].

ElecSim is made up of six fundamental components: 1) power plant data; 2) scenario data; 3) the time-steps of the algorithm; 4) the power exchange; 5) the investment algorithm and 6) the generation companies (GenCos) as agents. ElecSim uses a subset of representative days of electricity demand, solar irradiance and wind speed to approximate a full year. In this context,

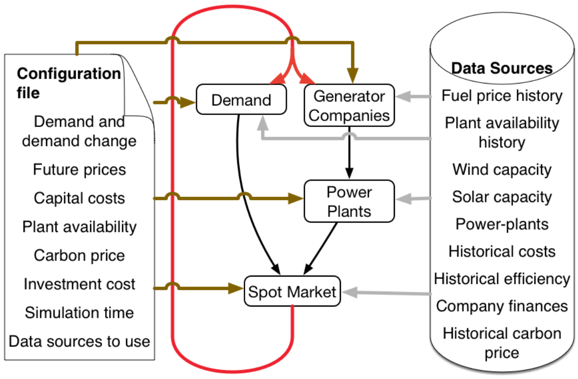


Fig. 5.14 System overview of ElecSim [101].

representative days are a subset of days which, when scaled up, represent an entire year [104]. We show how these components interact in Figure 5.14 [101]. Namely, electricity demand is matched with the supply provided by power plants through the use of a spot market. Generator companies invest in power plants based upon information provided by the data sources and expectation of the data provided by the configuration file.

ElecSim uses a configuration file which details the scenario which can be set by the user. This includes electricity demand, carbon price and fuel prices. The data sources parametrize the ElecSim simulation to make a digital twin of a particular country, including information such as wind capacity and power plants in operation. Generation Companies own and invest in power plants. These power plants are then matched to electricity demand using a spot market.

The market runs a merit-order dispatch model, and bids are made by the power plant's short-run marginal cost (SRMC). A merit-order dispatch model is one which dispatches the cheapest electricity generators first. SRMC is the cost it takes to dispatch a single MWh of electricity and does not include capital costs. Investment in power plants is based upon a net present value (NPV) calculation. NPV is the difference between the present value of cash inflows and the present value of cash outflows over a period of time. This is shown in Equation 6.1, where t is the year of the cash flow, i is the discount rate, N is the total number of years, or lifetime of power plant, and R_t is the net cash flow of the year t:

$$NPV(t,N) = \sum_{t=0}^{N} \frac{R_t}{(1+t)^t}. \tag{5.34}$$

Each of the Generator Companies (GenCos) estimate the yearly income for each prospective power plant by running a merit-order dispatch electricity market simulation ten years into the future. However, it is true that the expected cost of electricity ten years into the future is particularly challenging to predict. We, therefore, use a reference scenario projected by the UK Government Department for Business and Industrial Strategy (BEIS), and use the predicted costs of electricity calibrated by Kell *et al* [104, 42]. The agents predict the future carbon price by using a linear regression model.

5.18 Methods

5.18.1 Data preparation

Similarly to our previous work in [102], we selected a number of calendar attributes and demand data from the GB National Grid Status dataset provided by the electricity market settlement company Elexon, and the University of Sheffield [4]. This dataset contained data between the years 2011-2018 for the United Kingdom. The calendar attributes used as predictors to the models were hour, month, day of the week, day of the month and year. These attributes allow us to account for the periodicity of the data within each day, month and year.

It is also the case that electricity demand on a public holiday which falls on a weekday is dissimilar to load behaviours of ordinary weekdays [105]. We, therefore, marked each holiday day to allow the model to account for this.

As demand data is highly correlated with historical demand, we lagged the input demand data. In this context, the lagged data is where we provide data of previous time steps at the input. For example, for predicting $t+1$, we use n inputs: $t, t-1, t-2, \ldots, t-n$. This enabled us to take into account correlations on previous days, weeks and the previous month. Specifically, we used the previous 28 hours before the time step to be predicted for the previous 1st, 2nd, 7th and 30th day. We chose this as we believe that the previous two days were the most relevant to the day to be predicted, as well as the weekday of the previous week and the previous month. We chose the previous 28 hours to account for a full day, plus an additional 4 hours to account for the previous day's correlation with the day to be predicted. We could have increased the number of days provided to the algorithm. However, due to time and computational constraints, we used our previously described intuition for lagged data selection. The number of lagged inputs to trial increases exponentially with each additional day added, therefore making the problem intractable when also trialling such a high number of algorithms and hyperparameters.

In addition to this, we marked each of the days with their respective seven seasons. These seasons were defined by the National Grid Short Term Operating Reserve (STOR) Market Information Report [124]. These differ from the traditional four seasons by splitting autumn into two further seasons, and winter into three seasons. Finally, to predict a full 24-hours ahead, we used 24 different models, 1 for each hour of the day.

The data is standardized and normalized using min-max scaling between -1 and 1 before training and predicting with the model. This is due to the fact that the inputs such as day of the week, hour of day are significantly smaller than that of demand. Therefore, the demand will influence the result more due to its larger value. However, this does not necessarily mean that demand has greater predictive power.

5.18.2 Algorithm Tuning

To find the optimum hyperparameters, cross-validation is used. As this time-series data were correlated in the time-domain, we took the first six years of data (2011-2017) for training and tested on the remaining year of data (2017-2018).

Each machine learning algorithm has a different set of parameters to tune. To tune the parameters in this paper, we used a grid search method. Grid search is a brute force approach that trials each combination of parameters at our choosing; however, for our search space was small enough to make other approaches not worth the additional effort.

Tables 5.5 and 5.6 display each of the models and respective parameters that were used in the grid search. Table 5.5 shows the offline machine learning methods, whereas Table 5.6 displays the online machine learning methods. Each of the parameters within the columns "Values" are trialled with every other parameter.

Whilst there is room to increase the total number of parameters, due to the exponential nature of grid-search, we chose a smaller subset of hyperparameters, and a larger number of regressor types. Specifically, with neural networks, there is a possibility to extend the number of layers as well as the number of neurons, to use a technique called deep learning. Deep learning is a class of neural networks that use multiple layers to extract higher levels of features from the input. For this paper, however, we decided to trial a large number of different models, instead of a large number of different configurations for neural networks.

5.18.3 Prediction Residuals in ElecSim

Each of the previously mentioned models trialled will have a certain degree of errors. Prediction residuals are the difference between the estimated and actual values. We collect the prediction residuals to form a distribution for each of the models. We then trial 80 different closed-form distributions to see which of the distributions best fits the residuals from each of the models. These 80 distributions were chosen due to their implementation in scikit-learn [**?**].

Once each of the prediction residual distributions are fit with a sensible closed-form distribution, we sample from this new distribution and perturb the demand for the electricity market at each time step within ElecSim.

By perturbing the market by the residuals, we can observe what the effects are of incorrect predictions of demand in an electricity market using the long-term electricity market model, ElecSim. We are able to understand the differences that prediction residuals have on long-term investment decisions as well as generators utilized.

5.19 Results

In this Section, we detail the accuracy of the algorithms and statistical models to predict 24 hours ahead for the day-ahead market. In addition to this, we display the impact of the errors on electricity generation investment and electricity mix from the years 2018 to 2035 using the agent-based model ElecSim.

5.19.1 Offline Machine Learning

To generate these results, we use a training set to train the data, and a test set to see how well each algorithm performs on the testing data. That is, how well the algorithm can predict data it is

Regressor Type	Parameters	Values	Parameters	Values	Parameters	Values
Linear	N/A	N/A				
Lasso	N/A	N/A				
Elastic Net	N/A	N/A				
Least-Angle	N/A	N/A				
Extra Trees	# Estimators	[16, 32]				
Random Forest	# Estimators	[16, 32]				
AdaBoost	# Estimators	[16, 32]				
Gradient Boosting	# Estimators	[16, 32]	learning rate	[0.8, 1.0]		
Support Vector	Kernel	[linear, rbf]	C	[1, 10]	Gamma	[0.001, 0.0001]
Multilayer Perceptron	Activation function	[tanh, relu]	hidden layer sizes	[1, 50]	Alpha	[0.00005, 0.0005]
K-Neighbours	# Neighbours	[5, 20, 50]				

Table 5.5 Hyperparameters for offline machine learning regression algorithms

Regressor Type	Parameters	Values	Parameters	Values	Parameters	Values
Linear	N/A	N/A				
Box-Cox	Power	[0.1, 0.05, 0.01]				
Multilayer Perceptron	Hidden layer sizes	[(10, 50, 100), (10), (20), (50), (10, 50)]				
Passive Aggressive	C	[0.1, 1, 2]	Fit intercept?	[True, False]	Max iterations	[1, 10, 100, 1000]

Table 5.6 Hyperparameters for online machine learning regression algorithms

yet to see. In our case, the training data was from 2011 to 2017, and the testing data was from 2017 to 2018.

Figure 6.3 displays the mean absolute error of each of the offline statistical and machine learning models on a log scale. It can be seen that the different models have varying degrees of success. The least accurate models were linear regression, the multilayer perceptron (MLP) model and the Least Angle Regression (LARS). These all have mean absolute errors over 10,000MWh. This error would be prohibitively high in practice; the max tendered national grid reserve is 6,000MWh, while the average tendered national grid reserve is 2,000MWh [124].

A number of models perform well, with a low mean absolute error. These include the Lasso, gradient Boosting Regressor and K-neighbours regressor. The best model, similar to [102], was the decision tree-based model, Extra Trees Regressor, with a mean absolute error of $1,604$MWh. This level is well within the average national grid reserve of 2,000MWh.

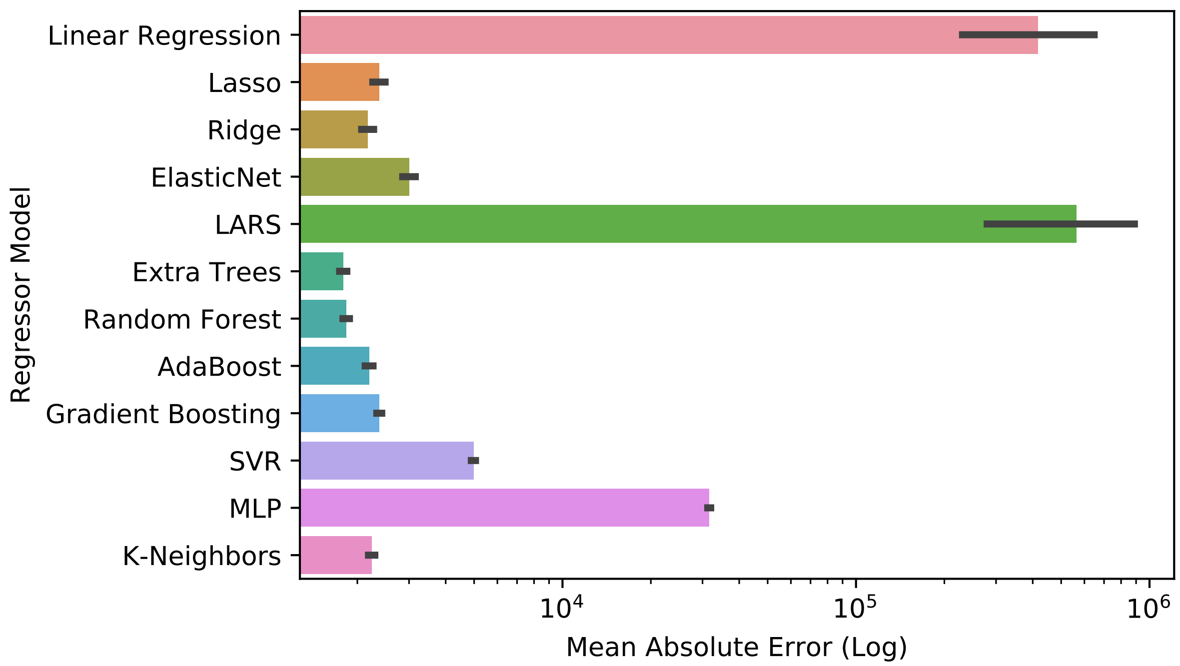


Fig. 5.15 Offline models mean absolute error comparison, with 95% confidence interval for 5 runs of each model.

Figure 5.16 displays the distribution of the best offline machine result (Extra Trees Regressor). It can be seen that the max tendered national grid reserve falls well above the 5% and 95% percentiles. However, there are occasions where the errors are greater than the maximum tendered national grid reserve. In addition, the majority of the time, the model's predictions fall within the average available tendered national grid reserve.

Figures 5.17 and 5.18 display the time taken to train the model and time taken to sample from the model versus the absolute error respectively for the offline algorithms. Multiple fits are trialled for each parameter type for each model. The error bars indicate the results of multiple cross-validations.

It can be seen from Figure 5.17 that the time to fit varies significantly between algorithms and parameter choices. The multilayer perceptron consistently takes a long time to fit, when

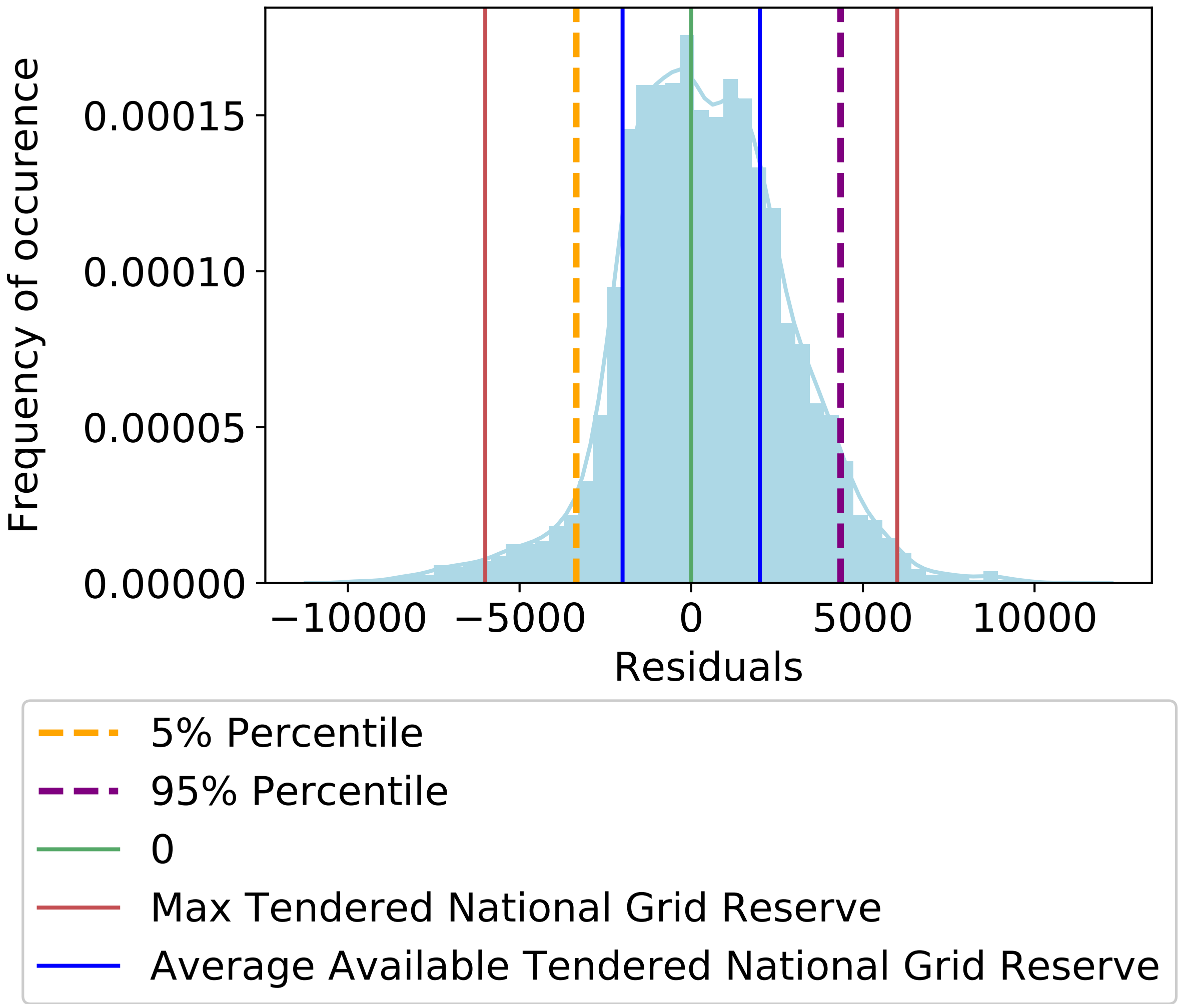


Fig. 5.16 Best offline machine learning algorithm (Extra Trees Regressor) distribution.

compared to the other algorithms and performs relatively poorly in terms of MAE. There are many models, such as the random forest regressor, and extra trees regressors which perform well, however, take a long time to fit, especially when compared to the K-Nearest neighbours.

For a small deterioration in MAE it is possible to decrease the time it takes to train the model significantly. For example, by using the K-Nearest neighbours or support vector regression (SVR).

The scoring time, displayed in Figure 5.18, also displays a large variation between model types. For instance, the MLP regressor takes a shorter time to sample predictions when compared to the K-Neighbors algorithm and support vector regression. It is possible to have a cluster of algorithms with low sample times and low mean absolute errors. However, often a trade-off is required, with a fast prediction time requiring a longer training time and vice-versa.

5.19.2 Online Machine Learning

To see if we can improve on the predictions, we utilize an online machine learning approach. If we are successful, we should be able to reduce the national grid reserves, reducing cost and emissions.

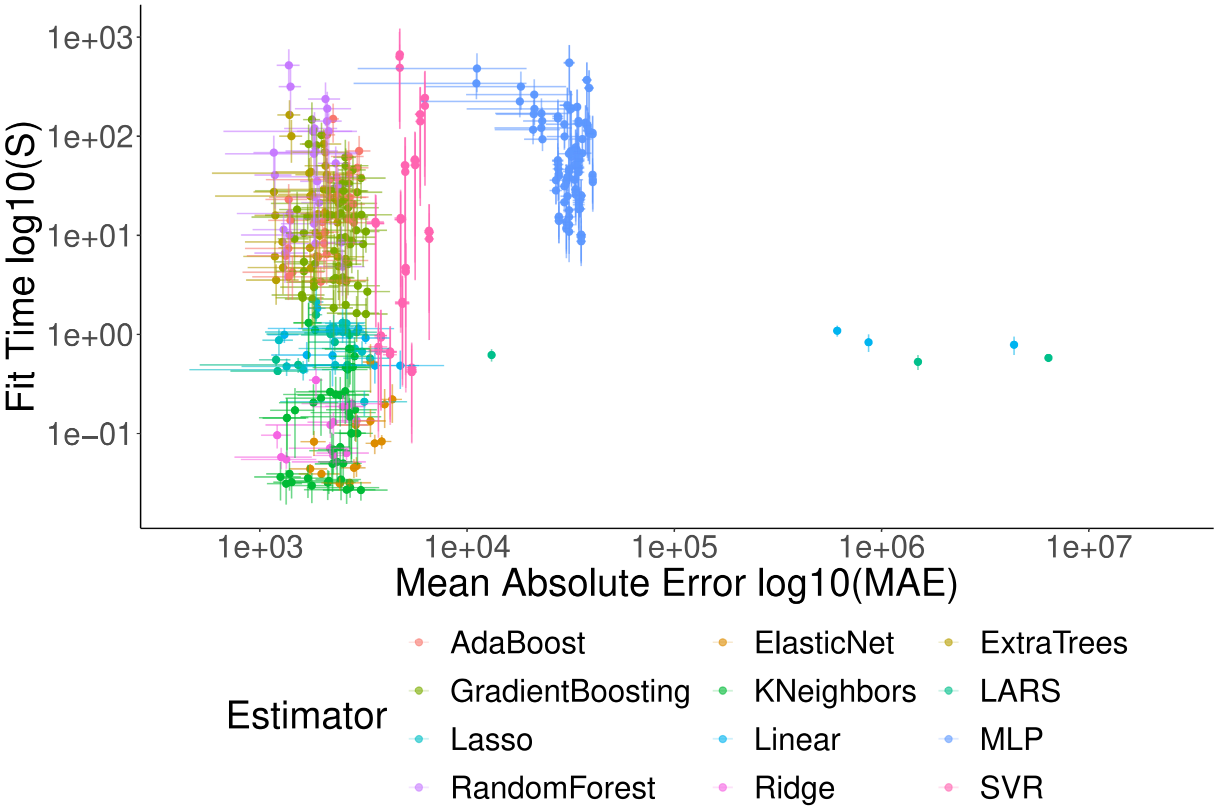


Fig. 5.17 Time taken to train the offline models versus mean absolute error. Error bars display standard deviation between points.

Figure 5.19 displays the comparison of mean absolute errors for the different trialled online regressor models. To produce this graph, we showed various hyperparameter trials. Where the hyperparameters had the same results, we removed them. For the multilayer perceptron (MLP), we aggregated all hyperparameters, due to the similar nature of the predictions.

It can be seen that the best performing model was the Box-Cox regressor, with an MAE of 1100. This is an improvement of over 30% on the best offline model. The other models perform less well. However, it can be seen that the linear regression model improves significantly for the online case when compared to the offline case. The passive aggressive (PA) model improve significantly with the varying parameters, and the MLP performs poorly in all cases.

Figure 5.20 displays the best online model. We can see a significant improvement over the best online model distribution, shown in Figure 5.16. We remain within the max tendered national grid reserve for 98.9% of the time, and the average available tendered national grid reserve is close to the 5% and 95% percentiles.

Figure 5.21 displays the residuals for a model with poor predictive ability, the passive aggressive regressor. It displays a large period of time of prediction errors at -20,000MWh, and often falls outside of the national grid reserve. These results demonstrate the importance of trying a multitude of different models and parameters to improve prediction accuracy.

Figure 5.22 displays a comparison between the actual electricity consumption compared to the predictions. It can be seen that the Box-Cox model better predicts the actual electricity demand in most cases when compared to the best offline model, the Extra Trees regressor. The Extra Trees regressor often overestimates the demand, particularly during weekdays. Whilst the

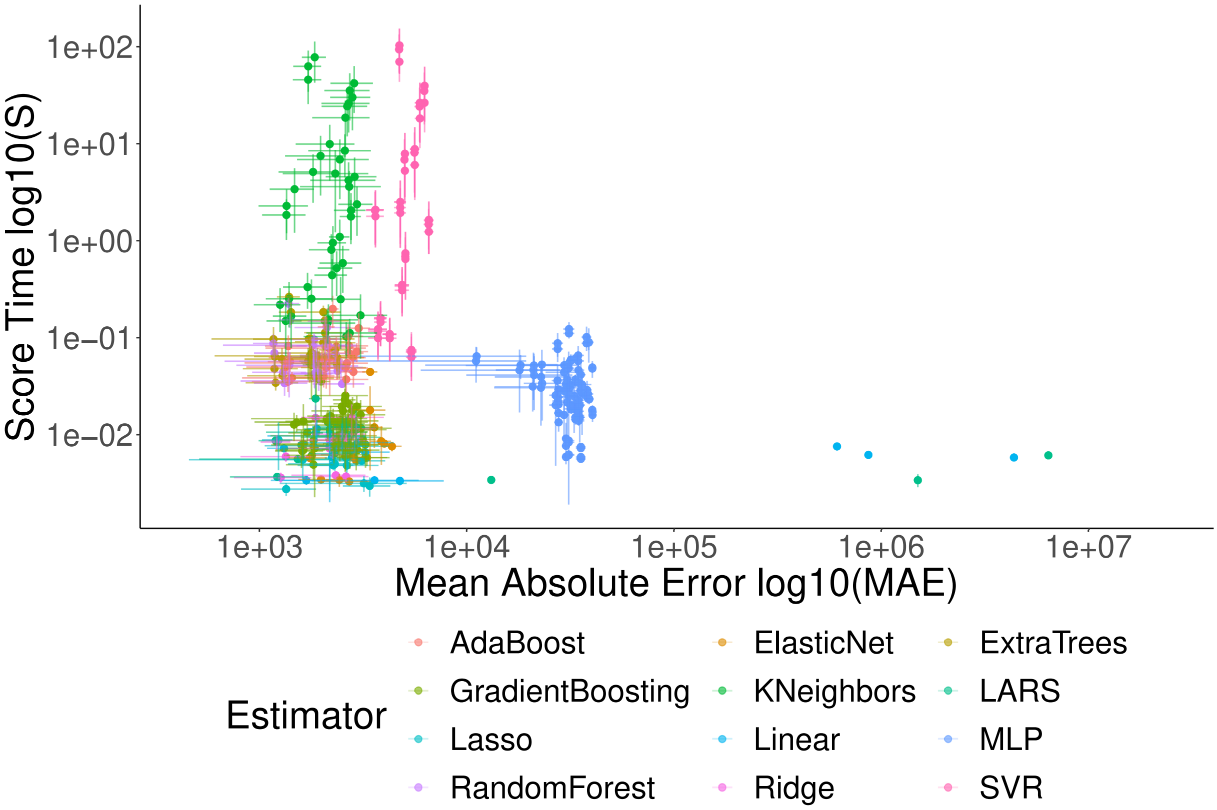


Fig. 5.18 Time taken to score the offline models versus mean absolute error. Error bars display standard deviation between points.

Box-Cox regressor more closely matches the actual results. During the weekend (between the hours of 120 and 168), the Extra Trees regressor performs better, particularly on the Saturday (between hours of 144 and 168).

Figures 5.23 and 5.24 display the mean absolute error versus test and training time respectively. In these graphs, a selection of models and parameter combinations are chosen.

Clear clusters can be seen between different types of models and parameter types. With the passive aggressive (PA) model performing the slowest for both training and testing. Different parameter combinations show different results in terms of mean absolute error.

The best performing model is the Box-Cox model, which is also the fastest to both train and test. The linear regression, which performs worse in terms of predictive performance, is as quick to train and test as the Box-Cox model. Additionally, the multilayer perceptron (MLP) is relatively quick to train and test when compared to the PA models.

It is noted that when compared to the offline models, the training time is a good indicator to the testing time. In other words, models that are fast to train are also fast to test and vice-versa.

5.19.3 Scenario Comparison

In this Section we explore the effect of these residuals on investments made and the electricity generation mix. To generate these graphs, we perturbed the exogenous demand in ElecSim by sampling from the best-fitting distributions for the respective residuals of each of the online

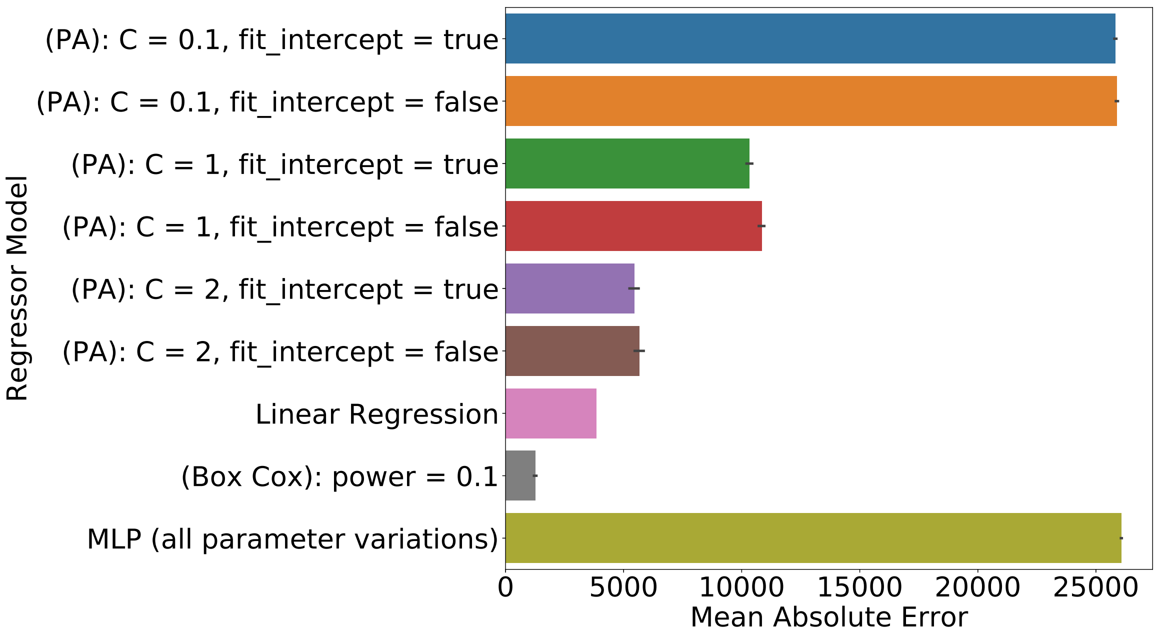


Fig. 5.19 Comparison of mean absolute errors (MAE) for different online regressor models. MLP results for all parameters are shown in a single barchart due to the very similar MAEs for the differing hyperparameters.

methods. We did this for all of the online learning algorithms displayed in Figure 5.19. We let the simulation run for 17 years from 2018 to 2035.

Running this simulation enabled us to see the effect on carbon emissions on the electricity grid over a long time period. For instance, does underestimating electricity demand mean that peaker power plants, such as reciprocal gas engines, are over utilized when other, less polluting power plants could be used?

Mean Contributed Energy Generation

In this Section we display the mean electricity mix contributed by different electricity sources over the years 2018 to 2035.

Figure 5.25a displays the mean photovoltaic (PV) contributed between 2018 and 2035 vs. mean absolute error of the various online regressor models displayed in Figure 5.19. A positive correlation can be seen with PV contributed and mean absolute error. This is similar for coal and nuclear output, shown in Figures 5.25b and 5.25c respectively. However, as shown by Figure 5.25d, offshore wind reduces with mean absolute error. Figure 5.25e displays the mean reciprocal gas engine output vs mean absolute error between the same time period. Output for the reciprocal gas engine also increases with mean absolute error.

The reciprocal gas engine was expected to increase with times of high error. This is because, traditionally, reciprocal gas engines are peaker power plants. Peaker power plants provide power at times of peak demand, which cannot be covered by other plants due to them being at their maximum capacity level or out of service. It may also be the case, that with higher proportions

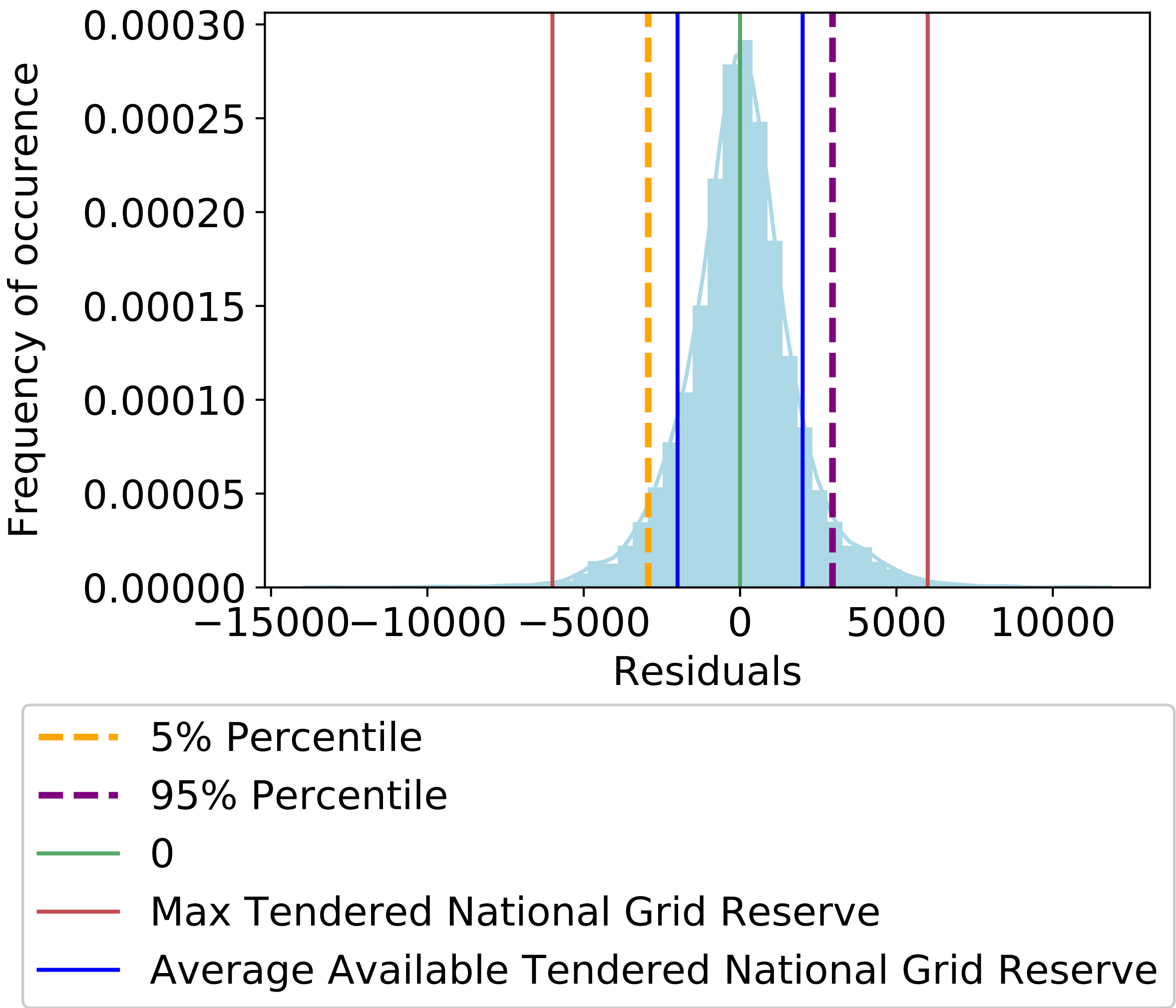


Fig. 5.20 Best online model (Box-Cox Regressor) distribution.

of intermittent technologies, there is a larger need for these peaker power plants to fill in for times where there is a deficit in wind speed and solar irradiance.

It is hypothesized that coal and nuclear output increase to cover the predicted increased demands of the service. As these generation types are dispatchable, meaning operators can choose when they generate electricity, they are more likely to be used in times of higher predicted demand.

Photovoltaics may be used more with higher errors due to the times at which the errors were greatest. For example, during the day, where demand is higher, as is solar irradiance.

Total Energy Generation

In this Section, we detail the difference in total technologies invested in over the time period between 2018 to 2035, as predicted by ElecSim.

CCGT, onshore, and reciprocal gas engines are invested in less over the time period, as shown by Figures 5.26a, 5.26d, 5.27a respectively. While coal, offshore, nuclear and photovoltaics all exhibit increasing investments.

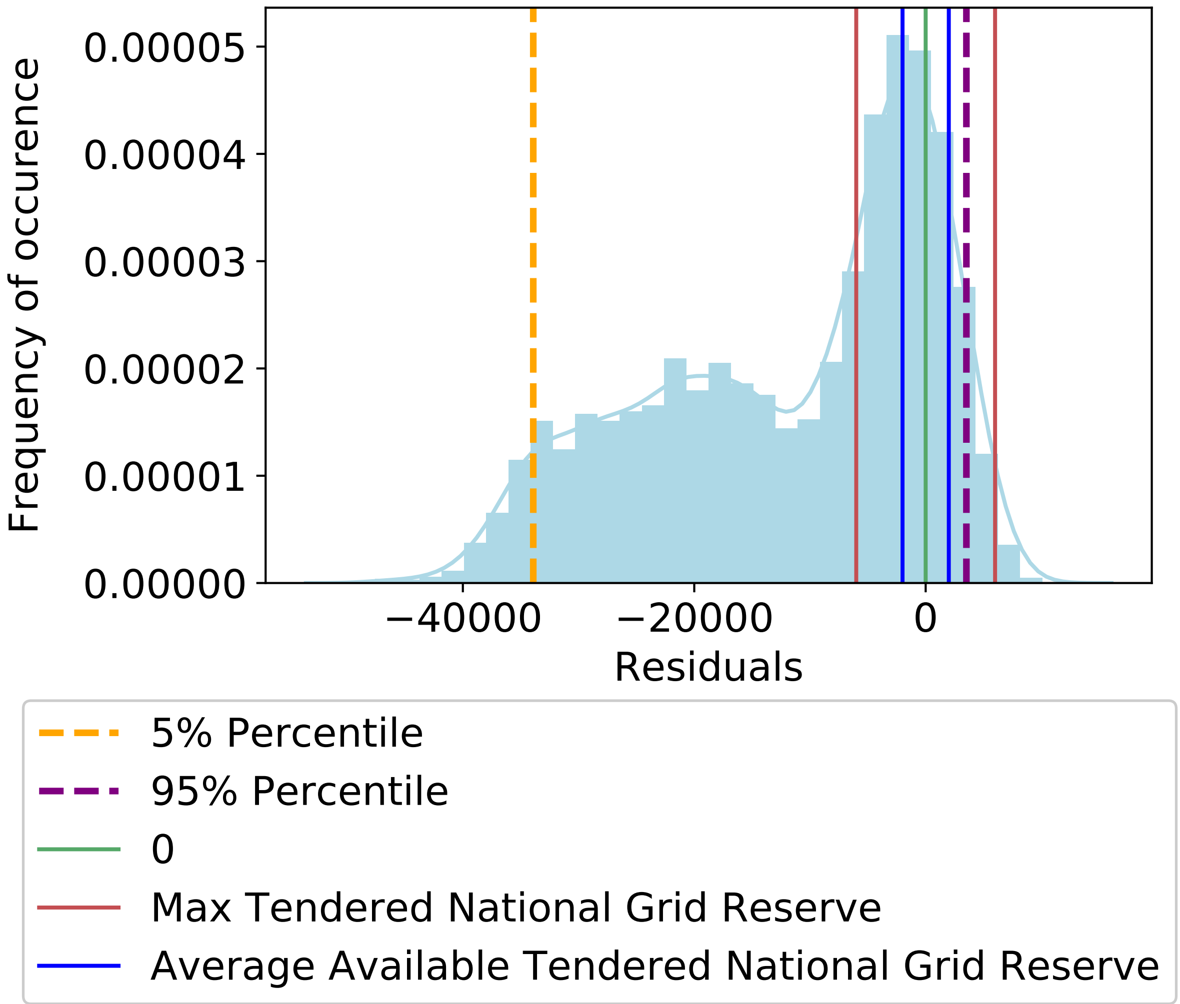


Fig. 5.21 Online machine learning algorithm distribution. (Passive Aggressive Regressor (C=0.1, fit intercept = true, maximum iterations = 1000, shuffle = false, tolerance = 0.001), chosen as it was the worst result for the passive aggressive model.

It is hypothesized that coal and nuclear increase in investment due to their dispatchable nature. While onshore, non-dispatchable by nature, become a less attractive investment when compared to the other technologies.

CCGT and reciprocal gas engines may have decreased in capacity over this time, due to the increase in coal. This could be because of the large consistent errors in prediction accuracy that meant that reciprocal gas engines were perceived to be less valuable.

Figure 5.27b shows an increase in relative mean carbon emitted with mean absolute error of the predictions residuals. The reason for an increase in relative carbon emitted could be due to the increased output of utility of the reciprocal gas engine, coal, and decrease in offshore output. Reciprocal gas engines are peaker plants and, along with coal, can be dispatched. By being dispatched, the errors in predictions of demand can be filled. It is therefore recommended that by improving the demand prediction algorithms, significant gains can be made in reducing carbon emissions.

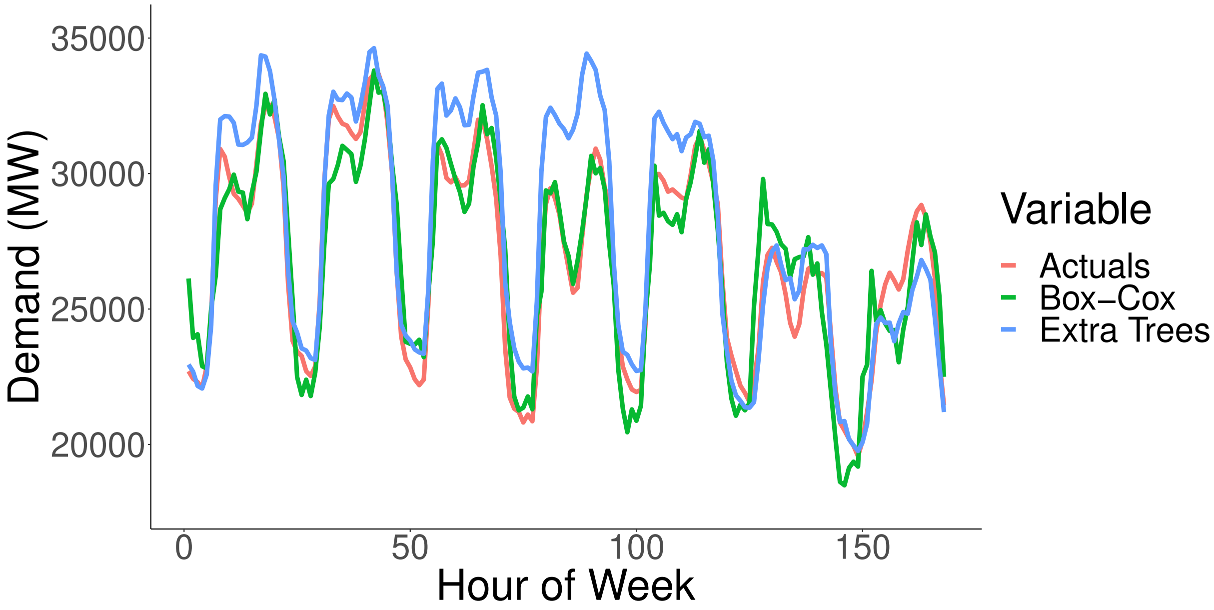


Fig. 5.22 Best offline model compared to the best online model over a one week period.

5.20 Discussion

From our results, it can be seen that different algorithms yield differing prediction accuracies. Online models can result in a decrease in 30% of prediction error on the best offline models. We calculated this by comparing the MAE for Extra Trees to the MAE for the Box-Cox regressor. We, therefore, recommend the use of online machine learning for predicting electricity demand in a day-ahead market.

Similar to our assumptions, the online learning algorithms were able to outperform the offline models. This is due to the non-stationary nature of the data. An online method is able to use the most up-to-date knowledge of the complex system of energy demand. For instance, a certain year may have a particularly warm winter when compared to previous years, reducing the amount of electricity used for heating.

However, contrary to our assumptions, the online linear regression techniques outperformed the online machine learning techniques. This may be due to their simpler nature and ability to learn from a smaller subset of new data as opposed to relying on a large historic subset. For the offline models, the best performing algorithms were the decision tree approaches such as extra trees and random forests. This is a similar outcome to our previous work, which showed that the best performing method for demand forecasting were random forests [103]. Contrary to our assumptions, however, the lasso and ridge regression outperformed the machine learning techniques support vector regression and multilayer perceptron. This may be due to the ability of feature selection by lasso and ridge regression, which only uses the most important features.

To the best of our knowledge, more work has been done using offline learning to predict electricity demand. This may be due to the additional complexity of running online algorithms, and a smaller number of available models to run in an online fashion.

In terms of computing power, finding the optimal input parameters, hyperparameters and models to use can be a large undertaking. This is due to the exponential growth of the number of

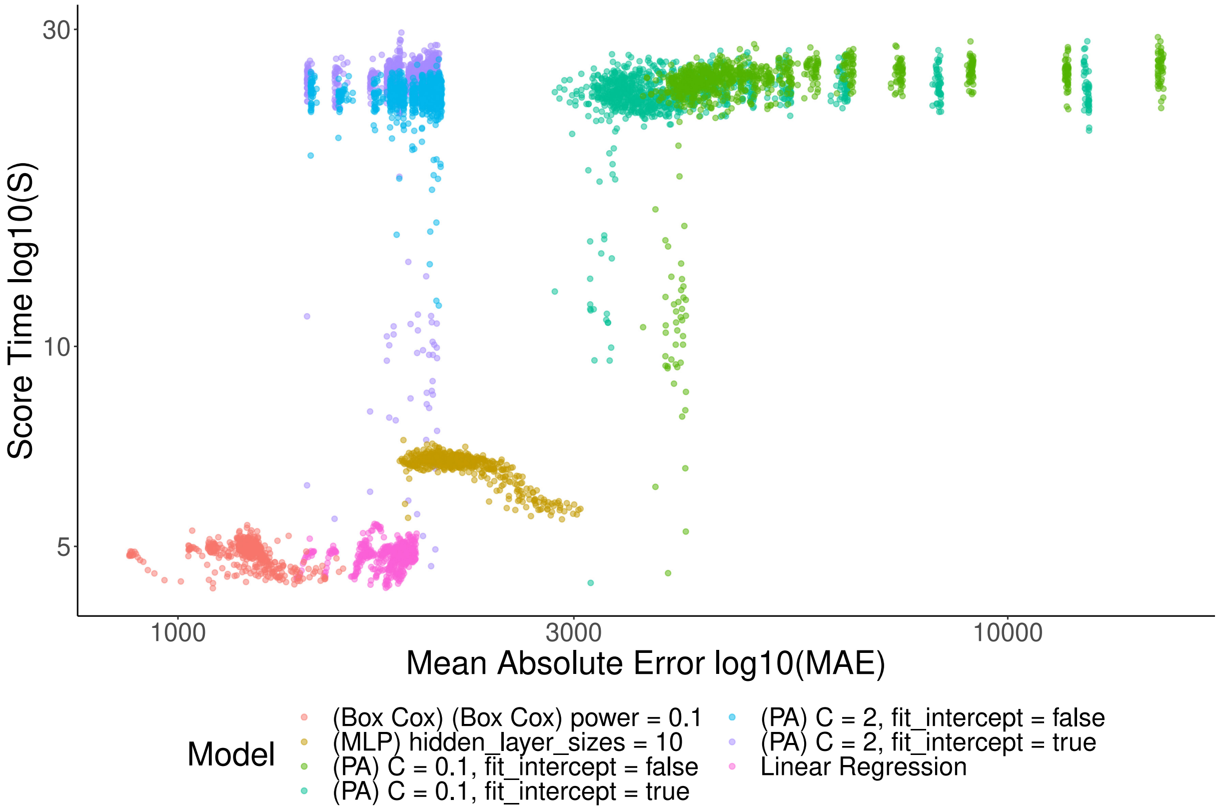


Fig. 5.23 Time taken to test the online models versus mean absolute error.

choices that can be made. This can be an issue where accuracy is of importance, especially when the data changes over time, meaning it may be necessary to retest previous results. However, due to the financial and sustainability implications, we believe the trade-off between compute time and accuracy is balanced towards compute time. There are also large implications if the model were to break at a certain point in time. We, therefore, recommend the reliance on multiple well-performing models, as opposed to solely the best performing model at any one time.

For training time and prediction time, there is often a trade-off between training and predicting. For instance, the k-nearest neighbours is fast to train, but slow to sample from. Therefore stakeholders must make a decision based upon accuracy, speed of training and sampling.

Additionally, the impact on the broader electricity market has been shown to be significant. Principally, the investment behaviours of generation companies change as well as the dispatched electricity mix. The relative mean carbon emitted over this time period increases, due to an increase in the utilization of coal and reciprocal gas engines, at the expense of offshore wind.

5.21 Conclusion

In this paper, we evaluated 16 different machine learning and statistical models to predict electricity demand in the UK for the day-ahead market. Specifically, we used both online and offline algorithms to predict electricity demand 24 hours ahead. We compared the ability for the offline models: lasso regression, random forests, support vector regression, for both online and offline learning: linear regression, multilayer perceptron and for just online learning: the Box-Cox transformation and the passive aggressive regressor, amongst others. The Box-Cox, as

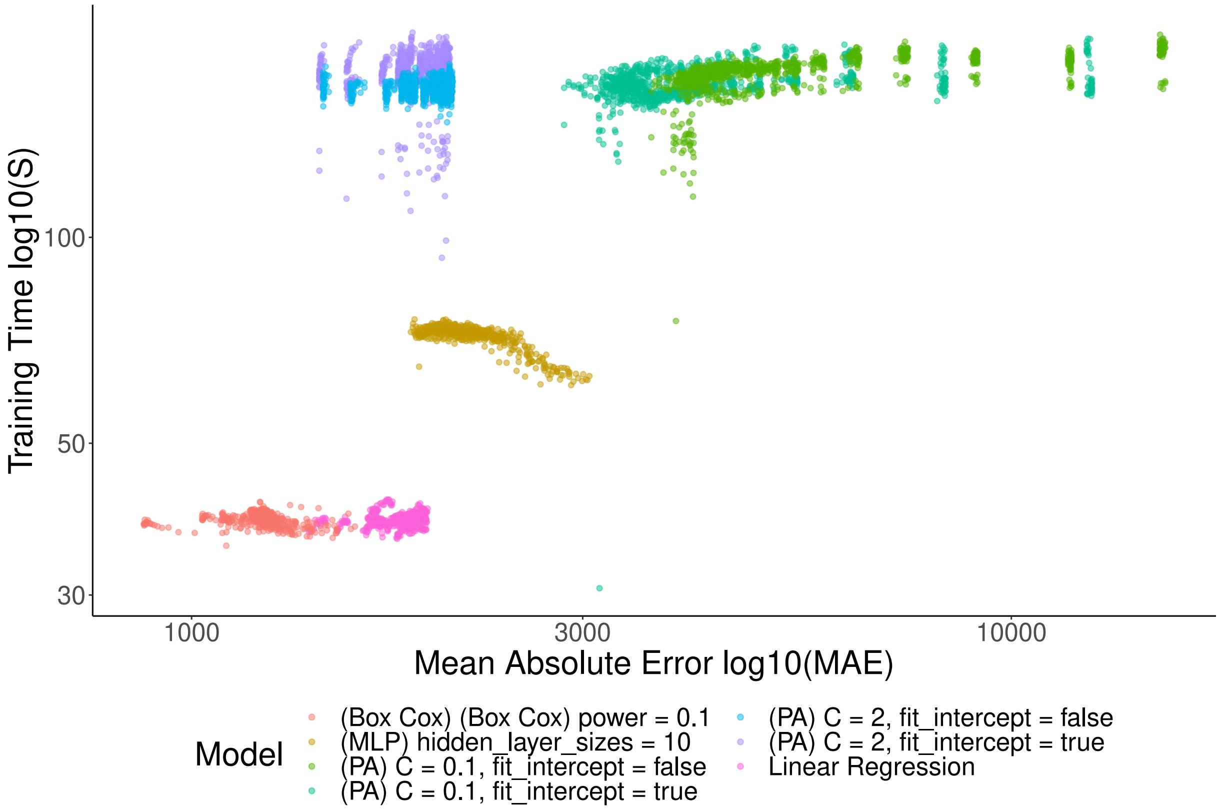
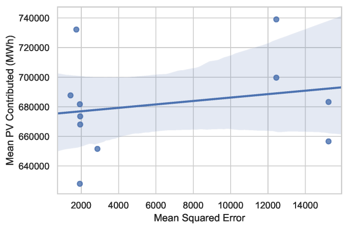
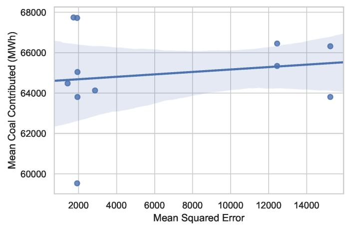


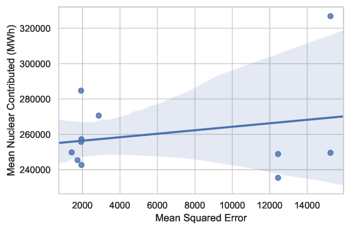
Fig. 5.24 Time taken to train the online models versus mean absolute error.



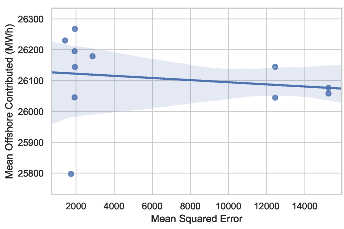
(a) Photovoltaic output.



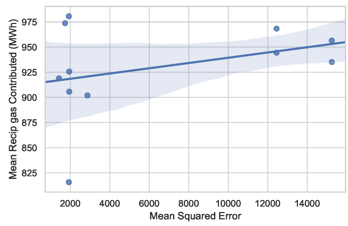
(b) Coal output.



(c) Nuclear output.



(d) Offshore output.

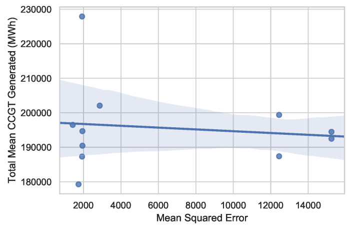


(e) Reciprocal gas engine output.

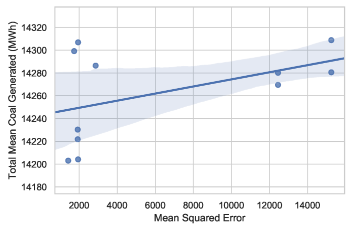
Fig. 5.25 Mean outputs of various technologies vs. mean absolute error from 2018 to 2035 in ElecSim.

well as the passive aggressive regressors, were used as online learning algorithms, the multilayer perceptron and linear regression were used as both, whereas the rest were used as offline learning algorithms.

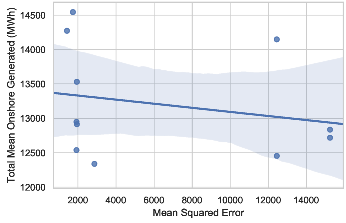
We measured the errors and compared these to each model as well as the national grid reserve. We found that through the use of an online learning approach, we were able to significantly



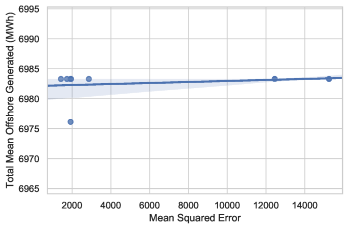
(a) Total CCGT.



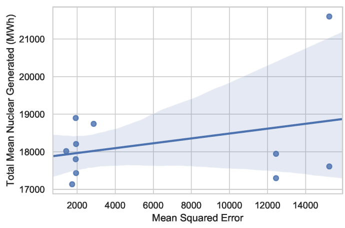
(b) Total Coal.



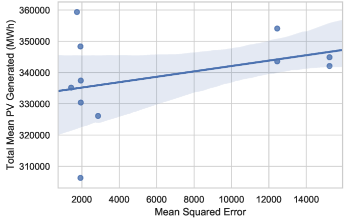
(c) Total Onshore.



(d) Total Offshore.

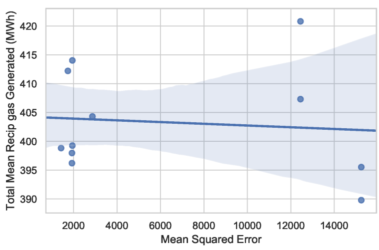


(e) Total nuclear.

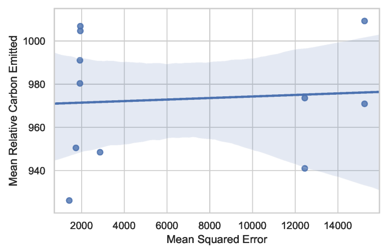


(f) Total photovoltaics.

Fig. 5.26 Total technologies invested in vs. mean absolute error from 2018 to 2035 in ElecSim.



(a) Total Reciprocal gas engine.



(b) Mean carbon emitted.

Fig. 5.27 a) Investments in reciprocal gas engine technologies vs. mean absolute error from 2018 to 2035 in ElecSim and d) mean carbon emissions between 2018 and 2035.

reduce error by 30% on the best offline algorithm. We were also able to reduce our errors to significantly below the national grid's mean and maximum tendered reserve, thus significantly reducing the chances of blackouts.

In addition to this, we took these errors, or residuals, and perturbed the electricity market of the agent-based model ElecSim. This enabled us to see the impact of different error distributions on the long-term electricity market, both in terms of investment and in terms of the electricity mix.

We observed that with an increase in prediction errors, we get a higher proportion of electricity generated by coal, offshore, nuclear, reciprocal gas engines and photovoltaics. This could be due to the fact that more peaker and dispatchable plants are required to fill in for unexpected demand. In addition, a higher proportion of intermittent renewable energy sources leads to a higher use of peaker power plants to fill in the gaps of intermittency of wind and solar irradiance. However, by reducing the mean absolute error, we are able to significantly reduce the amount of reciprocal gas engines and coal usage.

In future work, we would like to trial a different selection of algorithms and statistical models and trial different inputs to the models, for instance, by providing the model with two months worth of historical data as dependent variables. Additionally, we would like to see the impact of predicting wind speed and solar irradiance to see how these impact the overall investment patterns and electricity mix.

Chapter 6

Carbon optimization

6.1 Abstract

Electricity market modelling is often used by governments, industry and agencies to explore the development of scenarios over differing timeframes. For example, what would the reduction in cost of renewable energy mean for investments in gas power plants or what would be an optimum strategy for carbon tax or subsidies?

Optimization based solutions are the dominant approach for analysing energy policy. However, these types of models have certain limitations such as the need to be interpreted in a normative manner, and the assumption that the electricity market remains in equilibrium throughout. Through this work, we show that agent-based models are a viable technique to simulate decentralised electricity markets. The aim of this paper is to validate an agent-based modelling framework to increase confidence in its ability to be used in policy and decision making.

Our framework is able to model heterogenous agents with imperfect information. The model uses a rules-based approach to approximate the underlying dynamics of a real life, decentralised electricity market. We use the UK as a case-study, however our framework is generalisable to other countries. We increase the temporal granularity of the model by selecting representative days of electricity demand and weather using a *k*-means clustering approach.

We show that our modified framework, ElecSim, is able to adequately model the transition from coal to gas observed in the UK between 2013 and 2018. We are also able to simulate a future scenario to 2035 which is similar to the UK Government, Department for Business and Industrial Strategy (BEIS) predictions, showing a more realistic increase in nuclear power over this time period. This is due to the fact that with current, large nuclear technology, electricity is generated almost instantaneously and has a low relative short-run marginal cost [41]. This low short-run marginal cost means that new nuclear will be dispatched on the electricity market once it has been turned on, and thus will not increase gradually.

6.2 Introduction

Impacts on natural and human systems due to global warming have already been observed, with many land and ocean ecosystems having changed. A rise in carbon emissions increases the risk of severe impacts on the world such as rising sea levels, heat waves and tropical cyclones [120]. A study by Cook *et al.* demonstrated that 97% of scientific literature concurred that recent global warming was anthropogenic [34]. Limiting global warming requires limiting the total cumulative global anthropogenic emissions of CO_2 [120].

Global carbon emissions from fossil fuels, however, have significantly increased since 1900 [17]. Fossil-fuel based electricity generation sources such as coal and natural gas currently provide 65% of global electricity. Low-carbon sources such as solar, wind, hydro and nuclear provide 35% [20]. To halt this increase in CO_2 emissions, a transition of the energy system towards a renewable energy system is required.

However, such a transition needs to be performed in a gradual and non-disruptive manner. This ensures that there are no electricity shortages or power cuts that would cause damage to businesses, consumers and the economy.

To ensure such a transition, energy modelling is often used by governments, industry and agencies to explore possible scenarios under different variants of government policy, future electricity generation costs and energy demand. These energy modelling tools aim to mimic the behaviour of energy systems through different sets of equations and data sets to determine the energy interactions between different actors and the economy [117].

Optimization based solutions are the dominant approach for analysing energy policy [26]. However, the results of these models should be interpreted in a normative manner. For example, how investment and policy choices should be carried out, under certain assumptions and scenarios. The processes which emerge from an equilibrium model remain a black-box, making it difficult to fully understand the underlying dynamics of the model [26].

In addition to this, optimization models do not allow for endogenous behaviour to emerge from typical market movements, such as investment cycles [26, 65]. By modelling these naturally occurring behaviours, policy can be designed that is robust against movements away from the optimum/equilibrium. Thus, helping policy to become more effective in the real world.

The work presented in this paper builds on the agent-based model (ABM), ElecSim, developed by Kell *et al.* [101]. Agent-based models differ from optimization models by the fact that they are able to explore '*what-if*' questions regarding how a sector could develop under different prospective policies, as opposed to determining optimal trajectories. ABMs are particularly pertinent in decentralised electricity markets, where a centralised actor does not dictate investments made within the electricity sector. ABMs have the ability to closely mimic the real world by, for example, modelling irrational agents, in this case Generation Companies (GenCos) with incomplete information in uncertain situations [61].

There is a desire to validate the ability of energy-models to make long-term predictions. Validation increases confidence in the outputs of a model and leads to an increase in trust amongst the public and policy makers. Energy models, however, are frequently criticised for being insufficiently validated, with the performance of models rarely checked against historical outcomes [15].

In answer to this we postulate that ABMs can provide accurate information to decision makers in the context of electricity markets. We increase the temporal granularity of the work by Kell *et al.* [101] and use genetic algorithms to tune the model to observed data enabling us to perform validation. This enables us to understand the parameters required to observe certain phenomena, as well as use these fitted parameters to make inferences about the future.

We use a genetic algorithm approach to find an optimal set of price curves predicted by generation companies (GenCos) that adequately model observed investment behaviour in the real-life electricity market in the United Kingdom. Similar techniques can be employed for other countries of various sizes [101].

Similarly to Nahmmacher *et al.* we demonstrate how clustering of multiple relevant time series such as electricity demand, solar irradiance and wind speed can reduce computational time by selecting representative days [123]. In this context, representative days are a subset of days that have been chosen due to their ability to approximate the weather and electricity demand in an entire year. Distinct to Nahmacher *et al.* we use a k-means clustering approach [53] as opposed to a hierarchical clustering algorithm described by Ward [98]. We chose the k-means clustering approach due to previous success of this technique in clustering time series [102].

We measure the accuracy of projections for our improved ABM with those of the UK Government's Department for Business, Energy and Industrial Strategy (BEIS) for the UK electricity market between 2013 and 2018. In addition to this, we compare our projections from 2018 to 2035 to those made by BEIS in 2018 [42].

We are able to adequately model the transitional dynamics of the electricity mix in the United Kingdom between 2013 and 2018. During this time there was an $\sim$88% drop in coal use, $\sim$44% increase in Combined Cycle Gas Turbines (CCGT), $\sim$111% increase in wind energy and increase in solar from near zero to $\sim$1250MW. We are therefore able to test our model in a transition of sufficient magnitude.

We show in this paper that agent-based models are able to adequately mimic the behaviour of the UK electricity market under the same specific scenario conditions. Concretely, we show that under an observed carbon tax strategy, fuel price and electricity demand, the model ElecSim closely matches the observed electricity mix between 2013 and 2018. We achieve this by determining an exogenous predicted price duration curve using a genetic algorithm to minimise error between observed and simulated electricity mix in 2018. The predicted price curve is an arrangement of all price levels in descending order of magnitude. The predicted price duration curve achieved is similar to that of the simulated price duration curve in 2018, increasing confidence in the underlying dynamics of our model.

In addition, we compare our projections to those of the BEIS reference scenario from 2018 to 2035 [42]. To achieve this we use the same genetic algorithm optimisation technique as during our validation stage, optimising for predicted price duration curves. Our model demonstrates that we are able to closely match the projections of BEIS by finding a set of realistic price duration curves which are subject to investment cycles. Our model, however, exhibits a more realistic step change in nuclear output than that of BEIS. This is because, whilst BEIS projects a gradual increase in nuclear output, our model projects that nuclear output will grow instantaneously at a single point in time as a new nuclear power plant comes online.

This allows us to verify the scenarios of other models, in this case BEIS' reference scenario, by ascertaining whether the optimal parameters required to achieve such scenarios are realistic.

In addition to this, we are able to use these parameters to analyse '*what-if*' questions with further accuracy.

We increased the temporal granularity of the model using a k-means clustering approach to select a subset of representative days for wind speed, solar irradiance and electricity demand. This subset of representative days enabled us to approximate an entire year, and only required a fraction of the total time-steps that would be necessary to model each day of a year independently. This enabled us to decrease execution time. We show that we are able to provide an accurate framework, through this addition, to allow policy makers, decision makers and the public explore the effects of policy on investment in electricity generators.

We demonstrate that with a genetic algorithm approach we are able to optimise parameters to improve the accuracy of our model. Namely, we optimise the predicted electricity price, the uncertainty of this electricity price and nuclear subsidy. We use validation to verify our model using the observed electricity mix between 2013-2018.

The main contribution of this work is to demonstrate that it is possible for agent-based models to accurately model transitions in the UK electricity market. This was achieved by comparing our simulated electricity mix to the observed electricity mix between 2013 and 2018. In this time a transition from coal to natural gas was observed. We demonstrate that a high temporal granularity is required to accurately model fluctuations in wind and solar irradiance for intermittent renewable energy sources.

In Section 6.3 we introduce a review of techniques used for validating electricity market models as well as fundamental challenges of electricity model validation. Section 6.4 explores our approach to validate our model. In Section 6.5 we discuss the modifications made to our model to improve the results of validation. In Section 6.6 we present our results, and we conclude in Section 6.7.

6.3 Literature Review

In this section we cover the difficulties inherent in validating energy models and the approaches taken in the literature.

6.3.1 Limits of Validating Energy Models

Beckman *et al.* state that questions frequently arise as to how much faith one can put in energy model results. This is due to the fact that the performance of these models as a whole are rarely checked against historical outcomes [15].

Under the definition by Hodges *et al.* [80] long-range energy forecasts are not validatable [36]. Under this definition, validatable models must be observable, exhibit constancy of structure in time, exhibit constancy across variations in conditions not specified in the model and it must be possible to collect ample data [80].

Whilst it is possible to collect data for energy models, the data covering important characteristics of energy markets are not always measured. Furthermore, the behaviour of the human population and innovation are neither constant nor entirely predictable. This leads to the fact that static models cannot keep pace with global long-term evolution. Assumptions made by the modeller may be challenged in the form of unpredictable events, such as the oil shock of 1973 [36].

This, however, does not mean that energy-modelling is not useful for providing advice in the present. A model may fail at predicting the long-term future because it has forecast an undesirable event, which led to a pre-emptive change in human behaviour. Thus avoiding the original scenario that was predicted. This could, therefore, be viewed as a success of the model.

Schurr *et al.* argued against predicting too far ahead in energy modelling due to the uncertainties involved [156]. However, they specify that long-term energy forecasting is useful to provide basic information on energy consumption and availability which is helpful in public debate and in guiding policy makers.

Ascher concurs with this view and states that the most significant factor in model accuracy is the time horizon of the forecast; the more distant the forecast target the less accurate the model. This can be due to unforeseen changes in society as a whole [62].

It is for these reasons that we focus on a shorter-term (5-year) horizon window when validating our model. This enables us to have an increased confidence that the dynamics of the model work without external shocks and can provide descriptive advice to stakeholders. However, it must be noted that the UK electricity market exhibited a fundamental transition from natural gas to coal electricity generation during this period, meaning that a simple data-driven modelling approach would not work.

In addition to this short-term cross-validation, we compare our long-term projections to those of BEIS from 2018 to 2035. It is possible that our projections and those of BEIS could be wrong, however, this allows us to thoroughly test a particular scenario with different modelling approaches, and allow for the possibility to identify potential flaws in the models.

6.3.2 Validation Examples

In this section we explore a variety of approaches used in the literature for energy model validation.

The model OSeMOSYS [85] is validated against the similar model MARKAL/TIMES through the use of a case study named UTOPIA. UTOPIA is a simple test energy system bundled with ANSWER, a graphical user interface packaged with the MARKAL model generator [89, 126]. Hunter *et al.* use the same case study to validate their model Temoa [89]. In these cases, MARKAL/TIMES is seen as the "gold standard". In this paper, however, we argue that the ultimate gold standard should be real-world observations, as opposed to a hypothetical scenario.

The model PowerACE demonstrates that realistic prices are achieved by their modelling approach, however, they do not indicate success in modelling GenCo investment over a prolonged time period [136].

Barazza *et al.* validate their model, BRAIN-Energy, by comparing their results with a few years of historical data, however, they do not compare the simulated and observed electricity mix [13].

Work by Koomey *et al.* expresses the importance of conducting retrospective studies to help improve models [107]. In this case, a model can be rerun using historical data in order to determine how much of the error in the original forecast resulted from structural problems in the model itself, or how much of the error was due to incorrect specification of the fundamental drivers of the forecast [107].

A retrospective study published in 2002 by Craig *et al.* focused on the ability of forecasters to accurately predict electricity demand from the 1970s [36]. They found that actual energy usage in 2000 was at the very lowest end of the forecasts, with only a single exception. They found that these forecasts underestimated unmodelled shocks such as the oil crises which led to an increase in energy efficiency.

Hoffman *et al.* also developed a retrospective validation of a predecessor of the current MARKAL/TIMES model, named Reference Energy System [82], and the Brookhaven Energy System Optimization Model [2]. These were studies applied in the 70s and 80s to develop projections to the year 2000. This study found that the models had the ability to be descriptive, but were not entirely accurate in terms of predictive ability. They found that emergent behaviours in response to policy had a strong impact on forecasting accuracy. The study concluded that forecasts must be expressed in highly conditioned terms [81].

6.4 Problem Formulation

In this section we detail the approach taken in this paper to validate our model, including the parameters used for optimization.

Specifically, we use a genetic algorithm to find the predicted price duration curves which lead to the smallest error between our simulated electricity mix and the scenarios tested. The scenarios examined here are the observed electricity mix of the UK between 2013 and 2018 and the BEIS reference scenario projected in 2018 till 2035. When projecting the BEIS reference scenario we also optimise for nuclear subsidy and uncertainty in the price duration curves.

6.4.1 Optimization Variables

For GenCos to adequately make investments, they must formulate an expectation of future electricity prices over the lifetime of a plant. For this paper, we use the net present value (NPV) metric to compare investments.

NPV provides a method for evaluating and comparing investments with cash flows spread over many years, making it suited for evaluating power plants which have a long lifetime.

Equation 6.1 is the calculation of NPV, where t is the year of the cash flow, i is the discount rate, N is the total number of years, or lifetime of power plant, and R_t is the net cash flow of the year t.

The net cash flow, R_t, is calculated by subtracting both the operational and capital costs from revenue over the expected lifetime of the prospective plant. The revenue gained by each prospective plant is the expected price they will gain per expected quantity of MWh sold over the expected lifetime of the plant. This is shown formally in Equation 6.2:

$$NPV(i,N) = \sum_{t=0}^{N} \frac{R_t}{(1+t)^t} \tag{6.1}$$

$$R_t = \sum_{t=0}^{T}\sum_{h=0}^{H}\sum_{m=0}^{M} \left(m_{h,t}(PPDC_{h,t} - C_{var_{h,t}})\right) - C_c \tag{6.2}$$

where $m_{h,t}$ is the expected quantity of megawatts sold in hour h of year t. $PPDC_{h,t}$ is the predicted price duration curve at year t and hour h. $C_{var_{h,t}}$ is the variable cost of the power plant, which is dependent on expected megawatts of electricity produced, C_c is the capital cost.

The predicted price duration curve ($PPDC_{h,t}$) is an expectation of future electricity prices over the lifetime of the plant. The $PPDC_{h,t}$ is a function of supply and demand. However, with renewable electricity generator costs falling, future prices are uncertain and largely dependent upon long-term scenarios of electricity generator costs, fuel prices, carbon taxes and investment decisions. [92]. Due to the uncertainty of future electricity prices over the horizon of the lifetime of a power plant we have set future electricity prices as an exogenous variable that can be set by the user in ElecSim.

To gain an understanding of expected electricity prices that lead to particular scenarios we use a genetic algorithm optimisation approach. This enables us to understand the range of future electricity prices that lead to certain scenarios developing. In addition, it allows us to understand whether the parameters required for certain scenarios to develop are realistic. This enables us to check the assumptions of our model and the likelihood of scenarios.

Further, using these optimised parameters, we are better able to further explore '*what-if*' scenarios.

6.4.2 Validation with Observed Data

To verify the accuracy of the underlying dynamics of ElecSim, the model was initialised to data available in 2013 and allowed to develop until 2018. We used a genetic algorithm to find the optimum price duration curve predicted (*PPDC*) by the GenCos 10 years ahead of the year of the simulation. This *PPDC* was used to model expected rate of return of prospective generation types, as shown in Equations 6.1 and 6.2.

The genetic algorithm's objective was to reduce the error of simulated and observed electricity mix in the year 2018 by finding a suitable *PPDC* used by each of the GenCos for investment evaluation.

Scenario

For this experiment, we initialised ElecSim with parameters known in 2013 for the UK. ElecSim was initialised with every power plant and respective GenCo that was in operation in 2013 using the BEIS DUKES datatset [51]. The funds available to each of the GenCos was taken from publicly released official company accounts at the end of 2012 [40].

To ensure that the development of the electricity market from 2013 to 2018 was representative of the actual scenario between these years, we set the exogenous variables, such as carbon and fuel prices, to those that were observed during this time period. In other words, the scenario modelled equated to the observed scenario.

The data for the observed EU Emission Trading Scheme (ETS) price between 2013 and 2018 was taken from [140]. Fuel prices for each of the fuels were taken from [39]. The electricity load data was modelled using data from [gri], offshore and onshore wind and solar irradiance data was taken from [131]. There were three known significant coal plant retirements in 2016. These were removed from the simulation at the beginning of 2016.

Optimisation problem

The price duration curve was modelled linearly in the form $y = mx + c$, where y is the cost of electricity, m is the gradient, x is the demand of the price duration curve and c is the intercept.

Equation 6.3 details the optimisation problem formally:

$$\min_{m,c} \sum_{o \in O} \left(\frac{|A_o - f_o(m,c)|}{||O||} \right) \tag{6.3}$$

where $o \in O$ refers to the average percentage electricity mix during 2018 for wind (both offshore and onshore generation), nuclear, solar, CCGT, and coal, where O refers to the set of these values. A_o refers to observed electricity mix percentage for the respective generation type in 2018. $f_o(m,c)$ refers to the simulated electricity mix percentage for the respective generation type, also in 2018. The input parameters to the simulation are m and c from the linear *PPDC*, previously discussed, ie. $y = mx + c$. $||O||$ refers to the cardinality of the set.

6.4.3 Long-Term Scenario Analysis

In addition to verifying the ability for ElecSim to mimic observed investment behaviour over 5 years, we compared ElecSim's long-term behaviour to that of the UK Government's Department for Business, Energy and Industrial Strategy (BEIS) [42]. This scenario shows the projections of generation by technology for all power producers from 2018 to 2035 for the BEIS reference scenario. This is the same scenario as discussed in the next section, 6.4.3.

Scenario

We initialised the model to 2018 based on [101]. The scenario for development of fuel prices and carbon prices were matched to that of the BEIS reference scenario [42].

Optimisation problem

The optimisation approach taken was a similar process to that discussed in Sub-Section 6.4.2, namely using a genetic algorithm to find the optimum expected price duration curve. However, instead of using a single expected price duration curve for each of the agents for the entire simulation, we used a different expected price duration curve for each year, leading to 17 different curves. This enabled us to model the non-static dynamics of the electricity market over this extended time period.

In addition to optimising for multiple expected price duration curves, we optimised for a nuclear subsidy, S_n. Further, we optimised for the uncertainty in the expected price parameters m and c, named σ_m and σ_c respectively, where σ is the standard deviation in a normal distribution. m and c are the parameters for the predicted price duration curve, as previously defined, of the form $y = mx + c$.

This enabled us to model the different expectations of future price curves between the independent GenCos. The addition of a nuclear subsidy as a parameter is due to the likely requirement for Government to provide subsidies for new nuclear [150].

A modification was made to the reward algorithm for the long-term scenario case. Rather than using the discrepancy between observed and simulated electricity mix in the final year (2018) as the reward, a summation of the error metric for each simulated year was used. This is detailed formally in Equation 6.4:

$$\min_{m \in M, m \in C} \sum_{y \in Y} \sum_{o \in O} \left(\frac{\left| A_{y_o} - f_{y_o}(m_y, c_y) \right|}{||O||} \right) \tag{6.4}$$

where M and C are the sets of the 17 parameters of m_y and c_y respectively for each year, y. $y \in Y$ refers to each year between 2018 and 2035. m_y and c_y refer to the parameters for the predicted price duration curve, of the form $y = mx + c$ for the year y. A_{y_o} refers to the actual electricity mix percentage for the year y and generation type o. Finally, $f_{y_o}(m_y, c_y)$ refers to the simulated electricity mix percentage with the input parameters to the simulation of m and c for the year y.

6.5 Implementation details

In this section we discuss the changes made to Kell *et al.* to improve the results of validation [101]. Further, we introduce the genetic algorithm used to find the optimal parameter sets.

ElecSim is made up of six distinct sections: 1) power plant data; 2) scenario data; 3) the time-steps of the algorithm; 4) the power exchange; 5) the investment algorithm and 6) the generation companies (GenCos) as agents. ElecSim has been previously published [101], however, we have made amendments to the original work in the form of efficiency improvements to decrease compute time as well as increase the granularity of time-steps from yearly to representative days. Representative days, in this context, are a subset of days which when scaled up to 365 days can adequately represent a year.

In this paper, we initialised the model to a scenario of the United Kingdom as an example, however, the fundamental dynamics of the model remain the same for other decentralised electricity markets. In this section we detail the modifications we made to ElecSim for this paper. Further details of the design decisions of ElecSim are discussed in [101].

6.5.1 Representative Days

In previously published work, ElecSim modelled a single year as 20 time-steps for solar irradiance, onshore and offshore wind and electricity demand [101]. Similarly to findings of other authors, this relatively low number of time-steps led to an overestimation of the uptake of intermittent renewable energy resources (IRES) and an underestimation of flexible technologies [115, 72]. This is due to the fact that the full intermittent nature of renewable energy could not be accurately modelled in such a small number of time-steps.

To address this problem, whilst maintaining a tractable execution time, we approximated a single year as a subset of proportionally weighted, representative days. This enabled us to reduce computation time. Each representative day consisted of 24 equally separated time-steps, which model hours in a day. Hourly data was chosen, as this was the highest resolution of the dataset available for offshore and onshore wind and solar irradiance [131]. A lower resolution would allow us to model more days, however, we would lose accuracy in terms of the variability of the renewable energy sources.

Similarly to Nahmmacher *et al.* we used a clustering technique to split similar days of weather and electricity demand into separate groups. We then selected the historic day that was closest to the centre of the cluster, known as the medoid, as well as the average of the centre, known as the centroid [123]. Distinct to Nahmmacher, however, we used the *k*-means clustering algorithm [53] as opposed to the Ward's clustering algorithm [98]. This was due to the ability for the *k*-means algorithm to cluster time-series into relevant groups [103]. These days were scaled proportionaly to the number of days within their respective cluster to approximate a total of 365 days.

Equation 6.5 shows the series for a medoid or centroid, selected by the *k*-means algorithm:

$$P_h^{x,i} = \{P_1, P_2, \ldots, P_{24}\} \tag{6.5}$$

where $P_h^{x,i}$ is the medoid for series x, where $x \in X$ refers to offshore wind capacity factor, onshore wind capacity factor, solar capacity factor and electricity demand, h is the hour of the day and i is the respective cluster. $\{P_1, P_2, \ldots, P_{24}\}$ refers to the capacity values at each hour of the representative day.

We then calculated the weight of each cluster. This gave us a method of assigning the relative importance of each representative day when scaling the representative days up to a year. The weight is calculated by the proportion of days in each cluster. This gives us a method of determining how many days within a year are similar to the selected medoid or centroid. The calculation for the weight of each cluster is shown by Equation 6.6:

$$w_i = \frac{n_i}{||N||} \tag{6.6}$$

where w_i is the weight of cluster i, n_i is the number of days in cluster i, and $||N||$ is the total number of days that have been used for clustering.

The next step was to scale up the representative days to represent the duration curve of a full year. We achieved this by using the weight of each cluster, w_i, to increase the number of hours that each capacity factor contributed in a full year. Equation 6.7 details the scaling process to turn the medoid or centroid, shown in Equation 6.5, into a scaled day. Where $\widetilde{P}_h^{x,i}$ is the scaled day:

$$\widetilde{P}_h^{x,i} = \{P_{1w_i}, P_{2w_i}, \ldots, P_{24w_i}\} \tag{6.7}$$

Equation 6.7 effectively extends the length of the day proportional to the amount of days in the respective cluster.

Finally, each of the scaled representative days were concatenated to create the series used for the calculations which required the capacity factors and the respective number of hours that each capacity factor contributed to the year. Equation 6.8 displays the total time series for series x, where each scaled medoid is concatenated to produce an approximated time series, $\widetilde{P}^x$:

$$\widetilde{P}^x = \left(\widetilde{P}_h^{x,1}, \widetilde{P}_h^{x,2}, \ldots, \widetilde{P}_h^{x,||N||}\right) \tag{6.8}$$

the total number of hours in the approximated time series, $\widetilde{P}^x$, is equal to the number of hours in a day multiplied by the number of days in a year, which gives the total number of hours in a year ($24 \times 365 = 8760$), as shown by Equation 6.9:

$$\sum_{w \in W} \sum_{t=1}^{T=24} (w_i t) = 24 \times 365 = 8760 \tag{6.9}$$

where $w \in W$ is the set of clusters.

6.5.2 Error Metrics

To measure the validity of our approximation using representative days and also compare the optimum number of days, or clusters, we used a technique similar to Poncelet *et al.* [44, 132]. We trialled the number of clusters against three different metrics: correlation (CE_{av}), normalised root mean squared error ($nRMSE$) and relative energy error (REE_{av}).

REE_{av} is the average value over all the considered time series $\widetilde{P}^x \in \widetilde{P}$ compared to the observed average value of the set $P^x \in P$. Where $P^x \in P$ are the observed time series and $\widetilde{P}^x \in \widetilde{P}$ are the scaled, approximated time series using representative days. REE_{av} is shown formally by Equation 6.10:

$$REE_{av} = \frac{\sum_{P^x \in P} \left(\left| \frac{\sum_{t \in T} DC_{P_t^x} - \sum_{t \in T} \widetilde{DC}_{\widetilde{P}_t^x}}{\sum_{t \in T} DC_{P_t^x}} \right| \right)}{||P||} \tag{6.10}$$

where $DC_{P_t^x}$ is the duration curve for P^x and $DC_{\widetilde{P}_t^x}$ is the duration curve for $\widetilde{P}^x$. In this context, the duration curve can be constructed by sorting the capacity factor and electrical load data from high to low. The $x-$axis for the DC exhibits the proportion of time that each capacity factor represents. The approximation of the duration curve is represented in this text as $\widetilde{DC}_{\widetilde{P}^x}$.

$t \in T$ refers to a specific time step of the original time series. $\widetilde{DC}$ refers to the approximated duration curve for $\widetilde{P}^x$. Note that in this text $|\cdot|$ refers to the absolute value, and $||\cdot||$ refers to the cardinality of a set and $||P||$ refers to the total number of of considered time series.

Specifically, the sum of the observed values, P^x, and approximated values, $\widetilde{P}^x$, for all of the time series are summed. The proportional difference is found, which is summed for each of the different series, x, and divided by the number of series, to give REE_{av}.

Another requirement is for the distribution of load and capacity factors for the approximated series to correspond to the observed time series. It is crucial that we can account for both high and low levels of demand and capacity factor for IRES generation. This enables us to model for times where flexible generation capacity is required.

The distribution of values can be represented by the duration curve (*DC*) of the time series. Therefore, the average normalised root-mean-square error ($NRMSE_{av}$) between each *DC* is used as an additional metric. The $NRMSE_{av}$ is shown formally by Equation 6.11:

$$NRMSE_{av} = \frac{\sum_{P^x \in P} \left(\frac{\sqrt{\frac{1}{||T||} \cdot \sum_{t \in T} (DC_{P^x_t} - \widetilde{DC}_{\widetilde{P}^x_t})^2}}{max(DC_{P^x}) - min(DC_{P^x})} \right)}{||P||}. \tag{6.11}$$

Specifically, the difference between the approximated and observed duration curves for each time-step t is calculated. The average value is then taken of these differences. This average value is then normalised for the respective time series P^x. The average of these average normalised values for each time series are then taken to provide a single metric, $NRMSE_{av}$.

The final metric used is the correlation between the different time series. This is used due to the fact that wind and solar output influences the load within a single region, solar and wind output are correlated, as well as offshore and onshore wind levels within the UK. This is referred to as the average correlation error (CE_{av}) and shown formally by Equation 6.12:

$$CE_{av} = \frac{2}{||P|| \cdot (||P|| - 1)} \cdot \left(\sum_{p_i \in P} \sum_{p_j \in P, j > i} \left| corr_{p_i, p_j} - \widetilde{corr}_{p_i, p_j} \right| \right) \tag{6.12}$$

where $corr_{p1,p2}$ is the Pearson correlation coefficient between two time series $p_1, p_2 \in P$, shown by Equation 6.13. Here, $V_{p1,t}$ represents the value of time series p_1 at time step t:

$$corr_{p1,p2} = \frac{\sum_{t \in T} \left(\left(V_{p1,t} - \overline{V}_{p1} \right) \cdot \left(V_{p2,t} - \overline{V}_{p2} \right) \right)}{\sqrt{\sum_{t \in T} \left(V_{p1,t} - \overline{V}_{p1} \right)^2 \cdot \sum_{t \in T} \left(V_{p2,t} - \overline{V}_{p2} \right)^2}}. \tag{6.13}$$

6.5.3 Integrating higher temporal granularity

To integrate the additional temporal granularity of the model, extra time-steps were taken per year. The higher temporal granularity of the model enabled us to accurately model the hourly fluctuations in solar and wind which leads to more accurate expectations of the investment opportunities of these technologies [115, 72].

GenCos make bids at the beginning of every time-step and the Power Exchange matches demand with supply in merit-order dispatch using a uniform pricing market. An example of electricity mix in a single representative day is shown in Figure 6.1.

Figure 6.1 displays the high utilization of low marginal-cost generators such as nuclear, wind and photovoltaics. At hour 19, an increase in offshore wind leads to a direct decrease in CCGT. In contrast to this, a decrease in offshore and onshore between the hours of 8 and 12 lead to an increase in dispatch of coal and CCGT. One would expect this behaviour to prevent blackouts

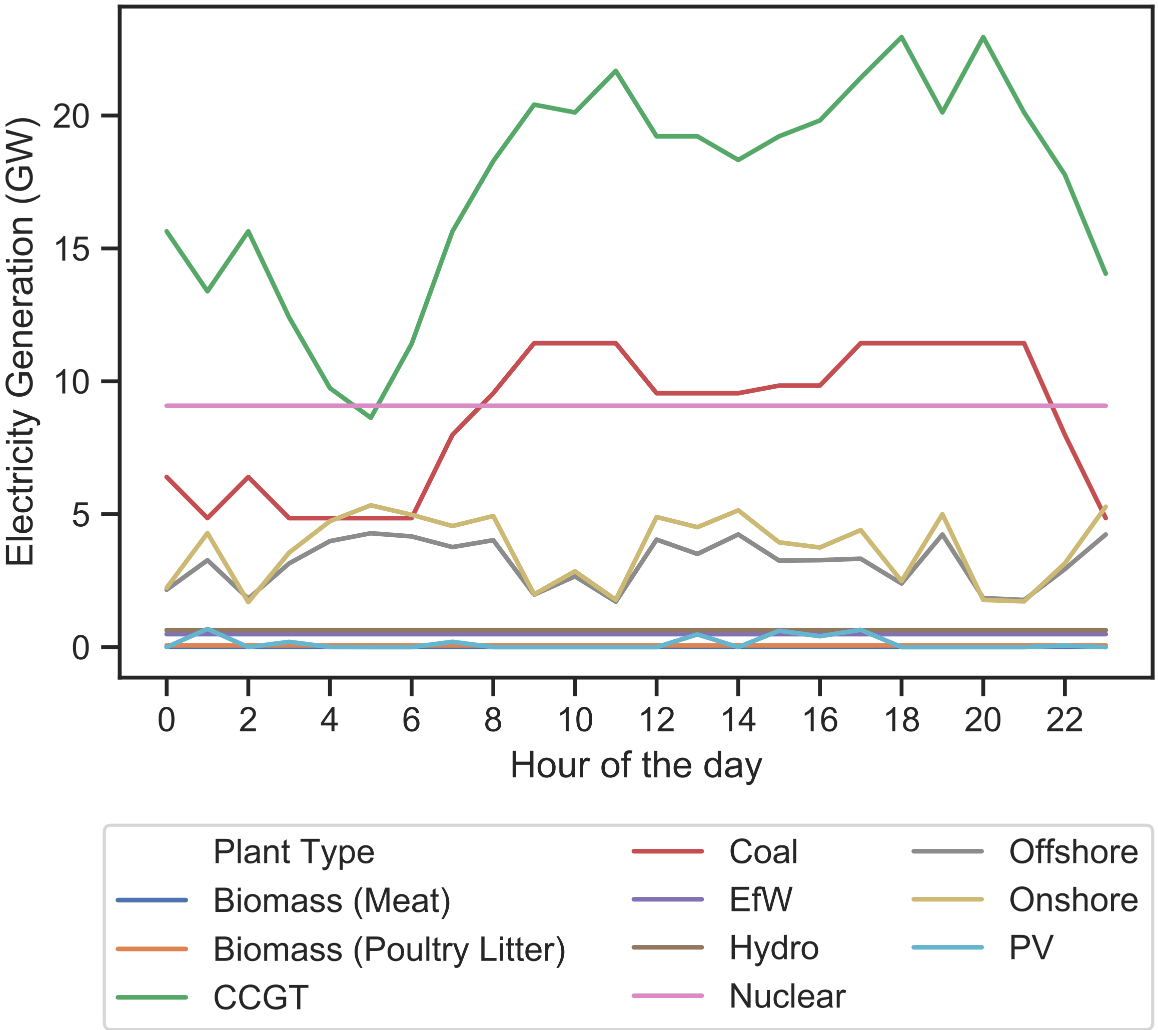


Fig. 6.1 Example of a single day of dispatched supply.

and meet demand at all times. This process has enabled us to more closely match fluctuations in IRES.

6.5.4 Genetic Algorithms

Genetic Algorithms (GAs) are a type of evolutionary algorithm which can be used for optimisation. We chose the genetic algorithm for this application due to its ability to find good solutions with a limited number of simulation runs, ability for parallel computation and its ability to find global optima. These characteristics are useful for our application, as a single simulation can take up to 36 hours.

In this section we detail the genetic algorithm used in this paper. Initially, a population P_0 is generated for generation 0. This population of individuals is used for the parameters to the simulation. The output of the simulations for each of the individuals are then evaluated. A subset of these individuals $C_{t+1} \subset P_t$ are chosen for mating. This subset is selected proportional to their fitness. With 'Fitter' individuals having a higher change of reproducing to create the

Algorithm 2 Genetic algorithm [11]

$t = 0$
initialize P_t
evaluate structures in P_t
while termination condition not satisfied **do**
    $t = t + 1$
    select reproduction C_t from P_{t-1}
    recombine and mutate structures in C_t forming C'_t
    evaluate structures in C'_t
    select each individual for P_t from C'_t or P_{t-1}
end while

offspring group C'_{t+1}. C'_{t+1} have characteristics dependent on the genetic operators: crossover and mutation. The genetic operators are an implementation decision [11].

Once the new population has been created, the new population P_{t+1} is created by merging individuals from C'_{t+1} and P_t. See Algorithm 2 for detailed pseudocode.

We used the DEAP evolutionary computation framework to create our genetic algorithm [55]. This framework gave us sufficient flexibility when designing our genetic algorithm. Specifically, it enabled us to persist the data of each generation after every iteration to allow us to verify and analyse our results in real-time.

Parameters for Validation with Observed Data

The parameters chosen for the problem explained in Section 6.4.2 was a population size of 120, a crossover probability of 50%, a mutation probability of 20% and the parameters, m and c, as per Equation 6.3, were given the bounds of $[0.0, 0.004]$ and $[-30, 100]$ respectively.

The bounds for m and c were calculated to ensure a positive price duration curve, with a maximum price of 300 for $50,000$MW. The population size was chosen to ensure a wide range of solutions could be explored, whilst limiting compute time to $\sim$1 day per generation to allow for sufficient verification of the results. The crossover and mutation probabilities were chosen due to suggestions from the DEAP evolutionary computation framework [55].

Parameters for Long-Term Scenario Analysis

The parameters chosen for the genetic algorithm for the problem discussed in Section 6.4.3 are displayed here. The population size was 127, a crossover probability of 50%, a mutation probability of 20%. The parameters m_y, c_y were given the bounds $[0.0, 0.003]$ and $[-30, 50]$ respectively, whilst σ_m and σ_c were both given the bounds of $[0, 0.001]$.

The population size was slightly increased, and the bounds reduced when compared to the parameters for Section 6.5.4. This was to increase the likelihood of convergence to a global optima, which was more challenging to achieve due to the significantly higher number of parameters.

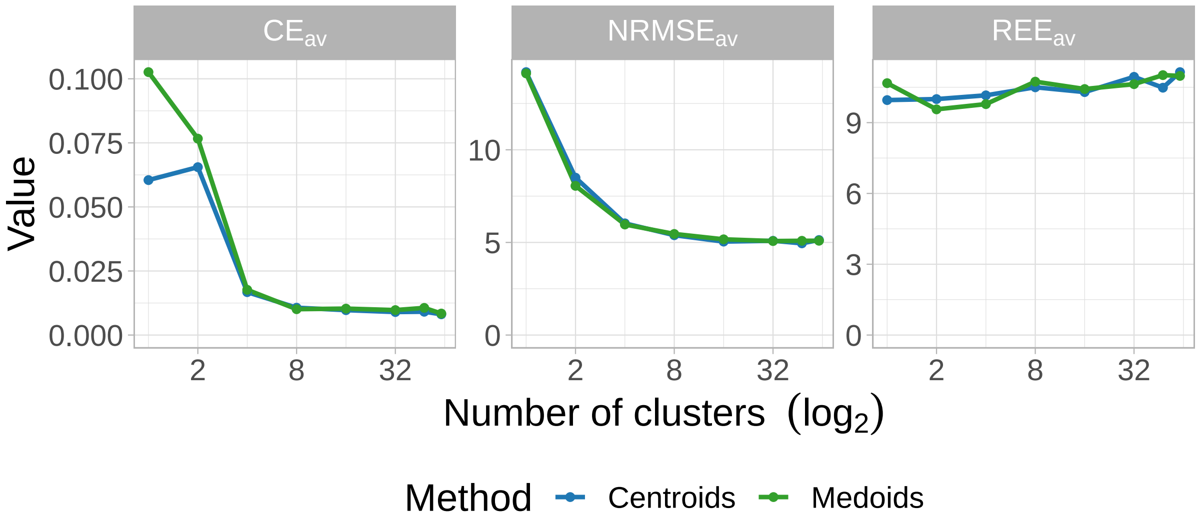


Fig. 6.2 Number of clusters compared to error metrics.

6.6 Results

Here we present the results of the problem formulation of Sections 6.4.2 and 6.4.3. Specifically, we compare the ability of our model to that of BEIS in the context of a historical validation between 2013 and 2018 of the UK electricity market. We also compare our ability to generate scenarios up to 2035 with that of BEIS.

6.6.1 Selecting representative days

Figure 6.2 displays the error metrics versus number of clusters. Both CE_{av} and $NRMSE_{av}$ display similar behaviour, namely the error improves significantly from a single cluster to eight clusters for both centroids and medoids. For the number of clusters greater than eight there are diminishing returns. For REE_{av}, however, the error metric is best at a single cluster, and gets worse with the number of clusters.

We chose eight clusters as a compromise between accuracy of the three error metrics and compute time of the simulation. This is because, eight was the largest number of clusters that gave us the lowest score for CE_{av}, $NRMSE_{av}$ and REE_{av} without significantly increasing compute time. Whilst there was little significant difference between centroid and medoid, we chose to use the medoids due to the fact that the extreme high and low values would not be lost due to averaging [75].

6.6.2 Validation with Observed Data

Figure 6.3 displays the output of ElecSim under the validation scenario, BEIS' projections and the observed electricity mix between 2013 and 2018, as explained in Sub-Section 6.4.2.

The observed electricity mix changed significantly between 2013 and 2018. A continuous decrease of electricity production from coal throughout this period was observed. 2015 and 2016 saw a marked decrease of coal, which can be explained by the retirement of 3 major coal power

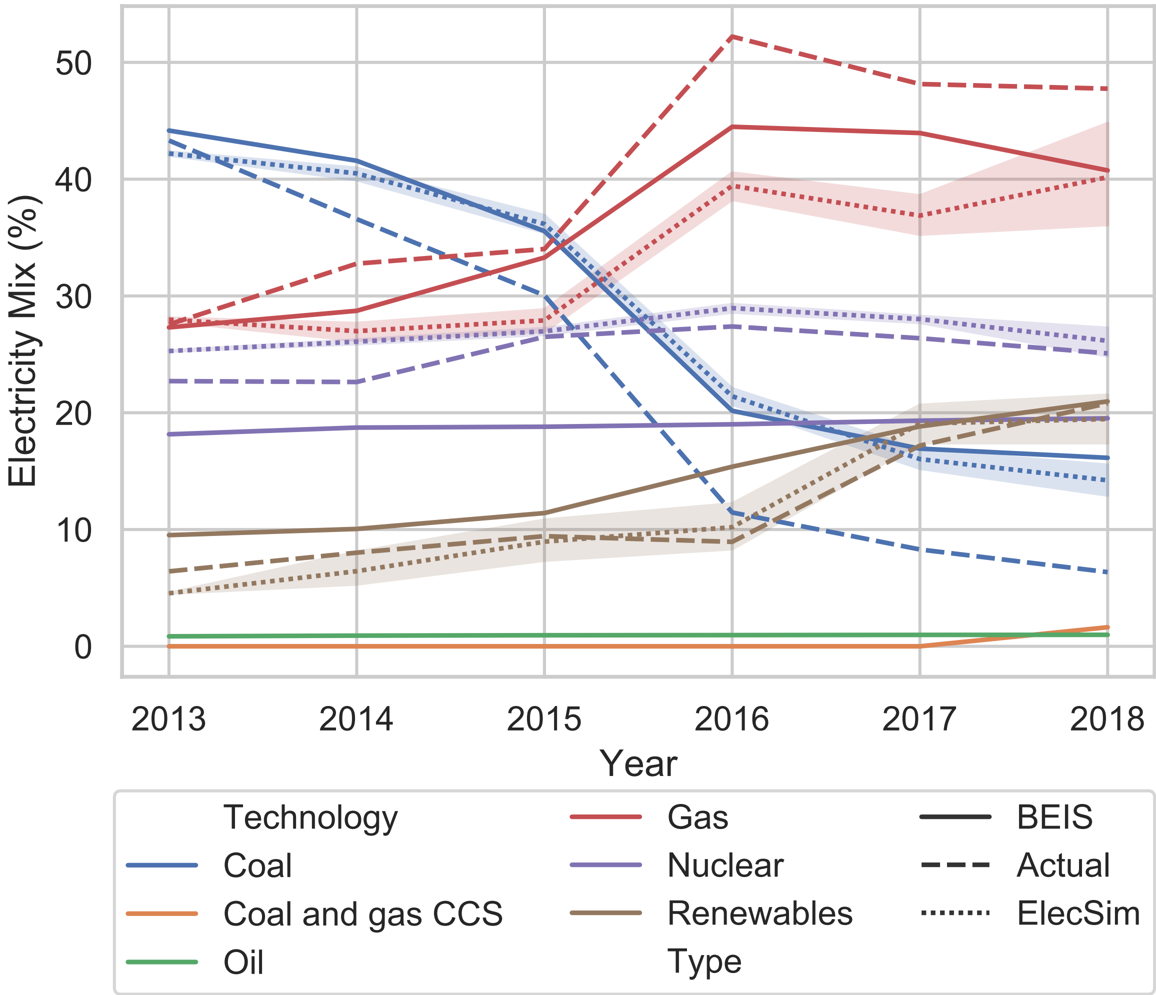


Fig. 6.3 Comparison of actual electricity mix vs. ElecSim vs. BEIS projections and taking three coal power plants out of service.

plants. The decrease in coal between 2013 and 2016 was largely replaced by an increase in gas. After 2016, renewables play an increasingly large role in the electricity mix and displace gas.

Both ElecSim and BEIS were able to model the fundamental dynamics of this shift from coal to gas as well as the increase in renewables. Both models, however, underestimated the magnitude of the shift from coal to gas. This could be due to unmodelled behaviours such as consumer sentiment towards highly polluting coal plants, a prediction from industry that gas would become more economically attractive in the future or a reaction to The Energy Act 2013 which aimed to close a number of coal power stations over the following two decades [127].

ElecSim was able to closely model the increase in renewables throughout the period in question, specifically predicting a dramatic increase in 2017. This is in contrast to BEIS who predicted that an increase in renewable energy would begin in 2016. However, both models were able to accurately predict the proportion of renewables in 2018.

ElecSim was able to better model the observed fluctuation in nuclear power in 2016. BEIS, on the other hand, projected a more consistent nuclear energy output. This small increase in nuclear power is likely due to the decrease in coal during that year. BEIS consistently underestimated the share of nuclear power.

We display the error metrics to evaluate our models 5 year projections in Table 6.1. Where MAE is mean absolute squared error, MASE is mean absolute scaled error and RMSE is root mean squared error.

We are able to improve the projections for all generation types when compared to the naive forecasting approach using ElecSim, as shown by the MASE. Where the naive approach is simply predicting the next time-step by using the last known time-step. In this case the last known time-step is the electricity mix percentage for each generation type in 2013.

Technology	MAE	MASE	RMSE
CCGT	9.007	0.701	10.805
Coal	8.739	0.423	10.167
Nuclear	1.69	0.694	2.002
Solar	0.624	0.419	1.019
Wind	1.406	0.361	1.498

Table 6.1 Error metrics for time series forecast from 2013 to 2018.

Figure 6.4 displays the optimal predicted price duration curve (*PPDC*) found by the genetic algorithm. This price curve was used by the GenCos to achieve the results shown in Figure 6.3.

The yellow points show the simulated price duration curve for the first year of the simulation (2018). The red line (Simulated Fit (2018)) is a linear regression that approximates the simulated price duration curve (PDC (2018)). The blue line shows the price duration curve predicted (*PPDC*) by the GenCos to be representative of the expected prices over the lifetime of the plant.

The optimal predicted price duration curve (*PPDC*) closely matches the simulated fit in 2018, shown by Figure 6.4. However, the *PPDC* has a slightly higher peak price and lower baseload price. This could be due to the fact that there is a predicted increase in the number of renewables with a low SRMC. However, due to the intermittency of renewables such as solar and wind, higher peak prices are required to generate in times of low wind and solar irradiance at the earth's surface.

To generate Figure 6.5, we ran 40 scenarios with the *PPDC* to observe the final, simulated electricity mix. The error bars are computed based on a Normal distribution 95% confidence interval.

ElecSim was able to model the increase in renewables and stability of nuclear energy in this time. ElecSim was also able to model the transition from coal to gas, however, underestimated the magnitude of the transition. This was similar to the projections BEIS made in 2013 as previously discussed.

6.6.3 Long-Term Scenario Analysis

In this section we discuss the results of the analysis of the BEIS reference scenario explained in Section 6.4.3. Specifically, we created a scenario that mimicked that of BEIS in ElecSim and optimised a number of parameters using a genetic algorithm to match this scenario. Through this we are able to gain confidence in the underlying dynamics of ElecSim to simulate long-term

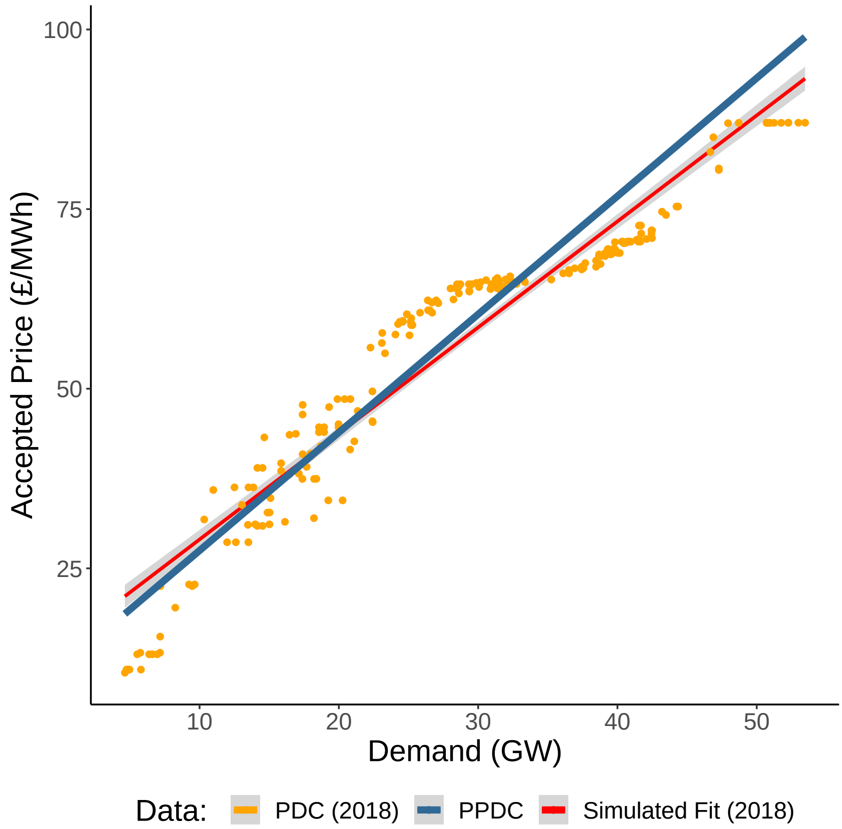


Fig. 6.4 Predicted price duration curve for investment for most accurate run against simulated run in 2018.

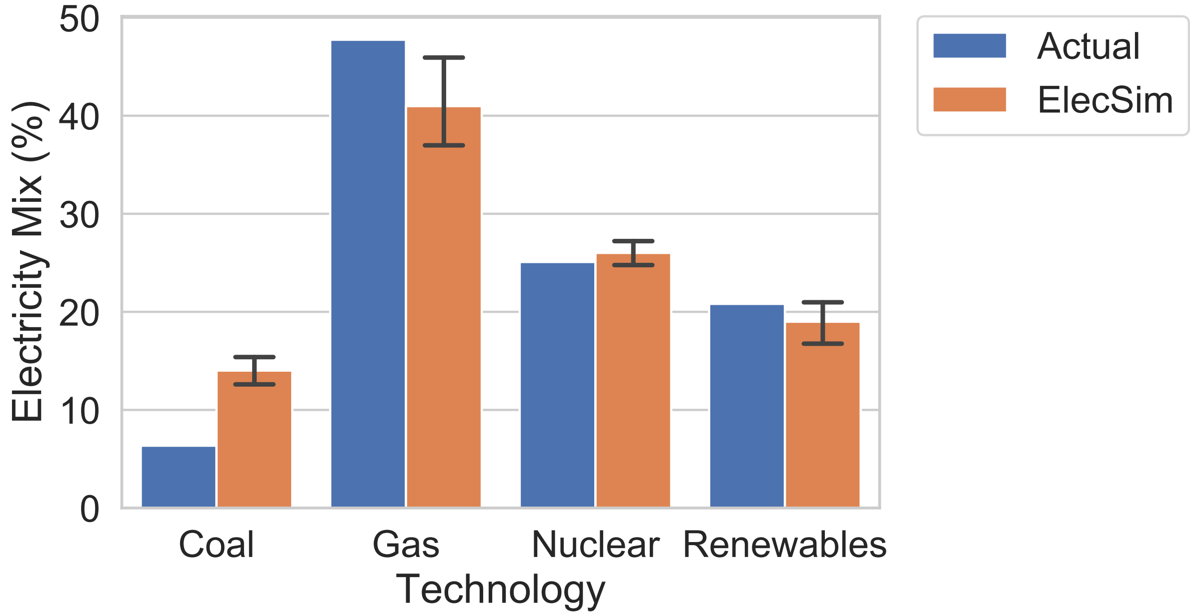


Fig. 6.5 Electricity generation mix simulated by ElecSim from 2013 to 2018 compared to observed electricity mix in 2018.

behaviours. Further, this enables us to verify the likelihood of the scenario by analysing whether the parameters required to make such a scenario are realistic.

Figure 6.6 displays the electricity mix projected by both ElecSim and BEIS. To generate this image we ran 60 scenarios under the optimal collection of predicted price duration curves, nuclear subsidy and uncertainty in predicted price duration curves.

The optimal parameters were chosen by choosing the parameter set with the lowest mean error per electricity generation type and per year throughout the simulation, as shown by Equation 6.4.

Figure 6.7 displays the optimal predicted price duration curves (*PPDC*s) per year of the simulation, shown in blue. These are compared to the price duration curve simulated in 2018, as per Figure 6.4. The optimal nuclear subsidy, S_n, was found to be $\sim$£120, the optimal σ_m and σ_c were found to be 0 and $\sim$0.0006 respectively.

The BEIS scenario demonstrates a progressive increase in nuclear energy from 2025 to 2035, a consistent decrease in electricity produced by natural gas, an increase in renewables and decrease to almost 0% by 2026 of coal.

ElecSim is largely able to mimic the scenario by BEIS. A large increase in renewables is projected, followed by a decrease in natural gas.

A significant difference, however, is the step-change in nuclear power in 2033. This led to an almost equal reduction in natural gas during the same year. In contrast, BEIS project a continuously increasing share of nuclear.

We argue that the ElecSim projection of nuclear power is more realistic than that of BEIS due to the instantaneous nature of large nuclear power plants coming on-line.

Figure 6.7 exhibits the price curves required to generate the scenario show in Figure 6.6. The majority of the price curves are similar to the simulated price duration curve of 2018 (Simulated Fit (2018)). However, there are some price curves which are significantly higher and significantly lower than the predicted price curve of 2018. These cycles in predicted price duration curves may be explained by investment cycles typically exhibited in electricity markets [65].

In this context, investment cycles reflect a boom and bust cycle over long timescales. When electricity supply becomes tight relative to demand, prices rise to create an incentive to invest in new capacity. Price behaviour in competitive markets can lead to periods of several years of low prices (close to short-run marginal cost) [161].

As plants retire or demand increases, the market becomes tighter until average prices increase to a level above the threshold for investment in new power generators. At this point investors may race to bring new plants on-line to make the most out of the higher prices. Once adequate investments have been made, the market returns to a period of low prices and low investment until the next price spike [65].

The nuclear subsidy, S_n, of $\sim$£120 in 2018 prices is high compared to similar subsidies, but this may reflect the difficulty of nuclear competing with renewable technology with a short-run marginal cost that tends to £0.

The low values of σ_m and σ_c demonstrates that the expectation of prices does not necessarily have to differ significantly between GenCos. This may be due to the fact that GenCos have access to the same market information.

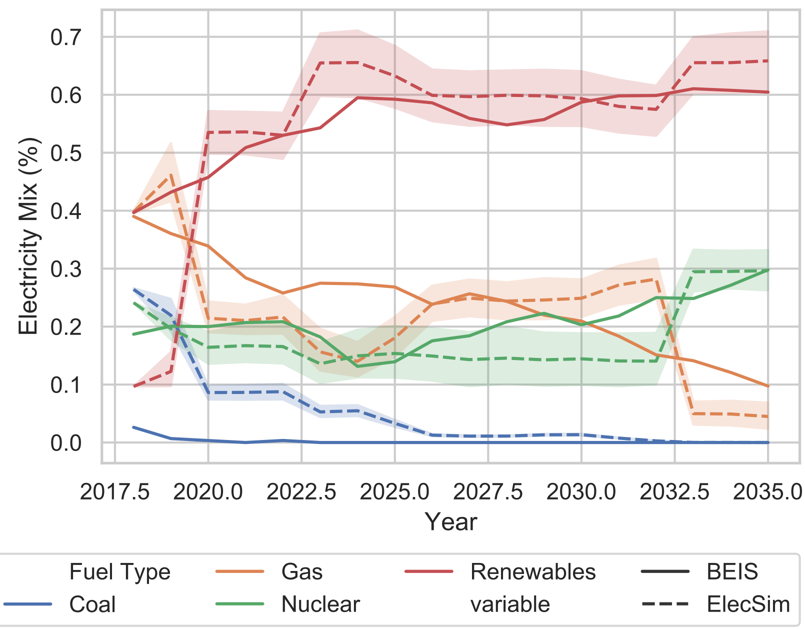


Fig. 6.6 Comparison of ElecSim and BEIS' reference scenario from 2018 to 2035.

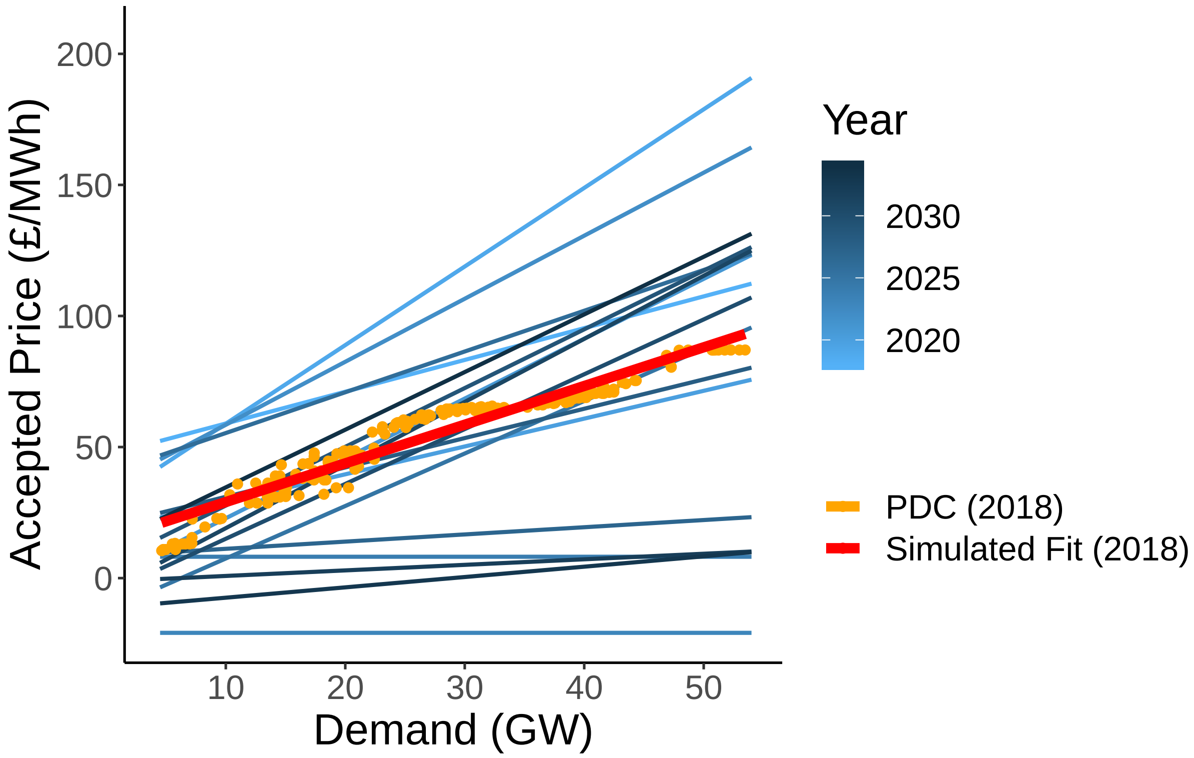


Fig. 6.7 Comparison between optimal price duration curves and simulated price duration curve in 2018.

6.7 Conclusion

In this paper we have demonstrated that it is possible to use agent based models to simulate liberalised electricity markets. Through validation we are able to show that our model, ElecSim, is able to accurately mimic the observed, real-life scenario in the UK between 2013 and 2018. This provides confidence in the underlying dynamics of ElecSim, especially as we are able to model the fundamental transition between coal and natural gas observed between 2013 and 2018 in the UK.

In addition to this, we were able to compare our long-term scenario to that of the UK Government, Department for Business, Energy & Industrial strategy. We show that we are able to mimic their reference scenario, however, demonstrate a more realistic increase in nuclear power. The parameters that were gained from optimisation show that the BEIS scenario is realistic, however a high nuclear subsidy may be required.

To improve the accuracy of our model, we used eight representative days of solar irradiance, offshore and onshore wind speed and demand to approximate an entire year. The particular days were chosen using a *k*-means clustering technique, and selecting the medoids. This enabled us to accurately model the daily fluctuations of demand and renewable energy resources.

In future work we would like to evaluate further scenarios to provide advice to stakeholders, integrate multi-agent reinforcement learning techniques to better model agents in both investment and bidding strategies as well as model different countries. Further work could be to make predicted price duration curves endogenous to the model, however, this could require scenario analysis by each of the GenCos each time they wanted to make an investment.

In addition to this, a method of dealing with the non-validatable nature of electricity markets, as per the definition of Hodges *et al.* is to vary input parameters over many simulations and look for general trends [80]. This could be achieved using ElecSim through the analysis of a reference case, and a limited set of scenarios which include the most important uncertainties in the model structure, parameters, and data, i.e. alternative scenarios which have both high plausibility and major impacts on the outcomes.

Chapter 7

Differential logistic equation models

Text goes here

Chapter 8

Conclusion

8.1 First section of the third chapter

And now I begin my third chapter here ...

And now to cite some more people **? ?**]

8.1.1 First subsection in the first section

... and some more

8.1.2 Second subsection in the first section

... and some more ...

First subsub section in the second subsection

... and some more in the first subsub section otherwise it all looks the same doesn't it? well we can add some text to it ...

8.1.3 Third subsection in the first section

... and some more ...

First subsub section in the third subsection

... and some more in the first subsub section otherwise it all looks the same doesn't it? well we can add some text to it and some more and some more and some more and some more and some more and some more and some more ...

Second subsub section in the third subsection

... and some more in the first subsub section otherwise it all looks the same doesn't it? well we can add some text to it ...

Table 8.1 A badly formatted table

	Species I		Species II	
Dental measurement	mean	SD	mean	SD
I1MD	6.23	0.91	5.2	0.7
I1LL	7.48	0.56	8.7	0.71
I2MD	3.99	0.63	4.22	0.54
I2LL	6.81	0.02	6.66	0.01
CMD	13.47	0.09	10.55	0.05
CBL	11.88	0.05	13.11	0.04

8.2 Second section of the third chapter

and here I write more ...

8.3 The layout of formal tables

This section has been modified from "Publication quality tables in LaTeX*" by Simon Fear.

The layout of a table has been established over centuries of experience and should only be altered in extraordinary circumstances.

When formatting a table, remember two simple guidelines at all times:

1. Never, ever use vertical rules (lines).

2. Never use double rules.

These guidelines may seem extreme but I have never found a good argument in favour of breaking them. For example, if you feel that the information in the left half of a table is so different from that on the right that it needs to be separated by a vertical line, then you should use two tables instead. Not everyone follows the second guideline:

There are three further guidelines worth mentioning here as they are generally not known outside the circle of professional typesetters and subeditors:

3. Put the units in the column heading (not in the body of the table).

4. Always precede a decimal point by a digit; thus 0.1 *not* just .1.

5. Do not use 'ditto' signs or any other such convention to repeat a previous value. In many circumstances a blank will serve just as well. If it won't, then repeat the value.

A frequently seen mistake is to use '\begin{center}' ... '\end{center}' inside a figure or table environment. This center environment can cause additional vertical space. If you want to avoid that just use '\centering'

Table 8.2 A nice looking table

Dental measurement	Species I		Species II	
	mean	SD	mean	SD
I1MD	6.23	0.91	5.2	0.7
I1LL	7.48	0.56	8.7	0.71
I2MD	3.99	0.63	4.22	0.54
I2LL	6.81	0.02	6.66	0.01
CMD	13.47	0.09	10.55	0.05
CBL	11.88	0.05	13.11	0.04

Table 8.3 Even better looking table using booktabs

Dental measurement	Species I		Species II	
	mean	SD	mean	SD
I1MD	6.23	0.91	5.2	0.7
I1LL	7.48	0.56	8.7	0.71
I2MD	3.99	0.63	4.22	0.54
I2LL	6.81	0.02	6.66	0.01
CMD	13.47	0.09	10.55	0.05
CBL	11.88	0.05	13.11	0.04

Chapter 9

My third chapter

Text goes here

Number 1

Windows OS

TeXLive package - full version

1. Download the TeXLive ISO (2.2GB) from
   https://www.tug.org/texlive/
2. Download WinCDEmu (if you don't have a virtual drive) from
   http://wincdemu.sysprogs.org/download/
3. To install Windows CD Emulator follow the instructions at
   http://wincdemu.sysprogs.org/tutorials/install/
4. Right click the iso and mount it using the WinCDEmu as shown in
   http://wincdemu.sysprogs.org/tutorials/mount/
5. Open your virtual drive and run setup.pl

or

Basic MikTeX - TEX distribution

1. Download Basic-MiKTEX(32bit or 64bit) from
   http://miktex.org/download
2. Run the installer
3. To add a new package go to Start » All Programs » MikTex » Maintenance (Admin) and choose Package Manager
4. Select or search for packages to install

TexStudio - TEX editor

1. Download TexStudio from
   http://texstudio.sourceforge.net/#downloads
2. Run the installer

Mac OS X

MacTeX - TEX distribution

1. Download the file from
   https://www.tug.org/mactex/
2. Extract and double click to run the installer. It does the entire configuration, sit back and relax.

TexStudio - TEX editor

1. Download TexStudio from
   http://texstudio.sourceforge.net/#downloads
2. Extract and Start

Unix/Linux

TeXLive - TEX distribution

Getting the distribution:

1. TexLive can be downloaded from
   http://www.tug.org/texlive/acquire-netinstall.html.
2. TexLive is provided by most operating system you can use (rpm,apt-get or yum) to get TexLive distributions

Installation

1. Mount the ISO file in the mnt directory

   ```
   mount -t iso9660 -o ro,loop,noauto /your/texlive####.iso /mnt
   ```

2. Install wget on your OS (use rpm, apt-get or yum install)
3. Run the installer script install-tl.

   ```
   cd /your/download/directory
   ./install-tl
   ```

4. Enter command 'i' for installation
5. Post-Installation configuration:
   http://www.tug.org/texlive/doc/texlive-en/texlive-en.html#x1-320003.4.1
6. Set the path for the directory of TexLive binaries in your .bashrc file

For 32bit OS

For Bourne-compatible shells such as bash, and using Intel x86 GNU/Linux and a default directory setup as an example, the file to edit might be

```
edit $~/.bashrc file and add following lines
PATH=/usr/local/texlive/2011/bin/i386-linux:$PATH;
export PATH
MANPATH=/usr/local/texlive/2011/texmf/doc/man:$MANPATH;
export MANPATH
INFOPATH=/usr/local/texlive/2011/texmf/doc/info:$INFOPATH;
export INFOPATH
```

For 64bit OS

```
edit $~/.bashrc file and add following lines
PATH=/usr/local/texlive/2011/bin/x86_64-linux:$PATH;
export PATH
MANPATH=/usr/local/texlive/2011/texmf/doc/man:$MANPATH;
export MANPATH
INFOPATH=/usr/local/texlive/2011/texmf/doc/info:$INFOPATH;
export INFOPATH
```

Fedora/RedHat/CentOS:

```
sudo yum install texlive
sudo yum install psutils
```

SUSE:

```
sudo zypper install texlive
```

Debian/Ubuntu:

```
sudo apt-get install texlive texlive-latex-extra
sudo apt-get install psutils
```

Number 2

LaTeX.cls files can be accessed system-wide when they are placed in the <texmf>/tex/latex directory, where <texmf> is the root directory of the user's TeXinstallation. On systems that have a local texmf tree (<texmflocal>), which may be named "texmf-local" or "localtexmf", it may be advisable to install packages in <texmflocal>, rather than <texmf> as the contents of the former, unlike that of the latter, are preserved after the LaTeXsystem is reinstalled and/or upgraded.

It is recommended that the user create a subdirectory <texmf>/tex/latex/CUED for all CUED related LaTeXclass and package files. On some LaTeXsystems, the directory look-up tables will need to be refreshed after making additions or deletions to the system files. For TeXLive systems this is accomplished via executing "texhash" as root. MIKTeXusers can run "initexmf -u" to accomplish the same thing.

Users not willing or able to install the files system-wide can install them in their personal directories, but will then have to provide the path (full or relative) in addition to the filename when referring to them in LaTeX.

References

[gri] Gb fuel type power generation production.

[2] (1975). National plan for energy research, development, and demonstration: Creating energy choices for the future.

[3] (2011). The irish electricity smart metering customer behaviour trials. [Online; accessed 21-January-2018].

[4] (2019).

[5] Abreu, J. M., Câmara Pereira, F., and Ferrão, P. (2012). Using pattern recognition to identify habitual behavior in residential electricity consumption. *Energy and Buildings*, 49:479–487.

[6] Ahmad, M. W., Mourshed, M., and Rezgui, Y. (2017). Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy and Buildings*, 147:77–89.

[7] Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

[8] Al-Musaylh, M., Deo, R., Adamowski, J., and Li, Y. (2018). Short-term electricity demand forecasting with MARS, SVR and ARIMA models using aggregated demand data in Queensland, Australia. *Advanced Engineering Informatics*, 35(November 2017):1–16.

[9] Alberta System Electric Operator (2016). AESO 2015 Annual Market Statistics. (March):28.

[10] Andersen, F. M., Larsen, H. V., and Boomsma, T. K. (2013). Long-term forecasting of hourly electricity load: Identification of consumption profiles and segmentation of customers. *Energy Conversion and Management*, 68:244–252.

[11] Back, T., Fogel, D. B., and Michalewicz, Z. (2009). Evolutionary Computation 1 Basic Algorithms and Operators. *Comprehensive Chemometrics*.

[12] Baram, Y., El-Yaniv, R., and Luz, K. (2003). Online Choice of Active Learning Algorithms. *Proceedings, Twentieth International Conference on Machine Learning*, 1:19–26.

[13] Barazza, E. and Strachan, N. (2020). The impact of heterogeneous market players with bounded-rationality on the electricity sector low-carbon transition. *Energy Policy*, 138(March 2019):111274.

[14] Batten, D. and Grozev, G. (2006). NEMSIM: Finding Ways to Reduce Greenhouse Gas Emissions Using Multi-Agent Electricity Modelling. *Complex Science for a Complex World*, pages 227–252.

[15] Beckman, J., Hertel, T., and Tyner, W. (2011). Validating energy-oriented CGE models. *Energy Economics*, 33(5):799–806.

[16] Benjamin, R. (2016). Tacit Collusion in Electricity Markets with Uncertain Demand. *Review of Industrial Organization*, 48(1):69–93.

[17] Boden, T., Andres, R., and Marland, G. (2017). Global, regional, and national fossil-fuel co2 emissions (1751-2014)(v. 2017).

[18] Böhringer, C. (1998). The synthesis of bottom-up and top-down in energy policy modeling. *Energy Economics*, 20(3):233–248.

[19] Box, G. E. P. and Cox, D. (1964). An Analysis of Transformations. *Journal, Source Statistical, Royal Series, Society*, 26(2):211–252.

[20] BP (2018). BP Statistical Review of World Energy. pages 1–56.

[21] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

[22] Breiman, L., Friedman, J., Stone, C., and Olshen, R. (1984). *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis.

[23] Carroll, J., May, A., McDonald, A., and McMillan, D. (2015). Availability Improvements from Condition Monitoring Systems and Performance Based Maintenance Contracts. (45):39.

[24] Chang, Y.-W., Hsieh, C.-J., Chang, K.-W., Ringgaard, M., and Lin, C.-J. (2010). Training and Testing Low-degree Polynomial Data Mappings via Linear SVM. *Journal of Machine Learning Research*, 11:1471–1490.

[25] Chaouch, M. (2014). Clustering-Based Improvement of Nonparametric Functional Time Series Forecasting : Application to. *IEEE Transactions on Smart Grid*, 5(1):411–419.

[26] Chappin, E. J., de Vries, L. J., Richstein, J. C., Bhagwat, P., Iychettira, K., and Khan, S. (2017). Simulating climate and energy policy with agent-based modelling: The Energy Modelling Laboratory (EMLab). *Environmental Modelling and Software*, 96:421–431.

[27] Chen, B.-j., Chang, M.-w., and Lin, C.-j. (2004). Load Forecasting Using Support Vector Machines : A Study on EUNITE Competition 2001. *IEEE Transactions on Power Systems*, 19(4):1821–1830.

[28] Chen, C., Tzeng, Y., and Hwang, J. (1996). The application of artificial neural networks to substation load forecasting. *Electric Power Systems Research*, 38(1996):153–160.

[29] Chen, L.-J., Zhu, L., Fan, Y., and Cai, S.-H. (2014). Long-Term Impacts of Carbon Tax and Feed-in Tariff Policies on China's Generating Portfolio and Carbon Emissions: A Multi-Agent-Based Analysis. *Energy & Environment*, 24(7-8):1271–1293.

[30] Cincotti, S., Gallo, G., and Berkeley, L. (2013). The Genoa Artificial Power-Exchange The Genoa artificial power-exchange. (January).

[31] Clastres, C. (2011). Smart grids: Another step towards competition, energy security and climate change objectives. *Energy Policy*, 39(9):5399–5408.

[32] Collins, S., Deane, J. P., Poncelet, K., Panos, E., Pietzcker, R. C., Delarue, E., and Ó Gallachóir, B. P. (2017). Integrating short term variations of the power system into integrated energy system models: A methodological review. *Renewable and Sustainable Energy Reviews*, 76(January):839–856.

[Conzelmann et al.] Conzelmann, G., Boyd, G., Koritarov, V., and Veselka, T. Multi-agent power market simulation using EMCAS. *IEEE Power Engineering Society General Meeting, 2005*, pages 917–922.

[34] Cook, J., Nuccitelli, D., Green, S. A., Richardson, M., Winkler, B., Painting, R., Way, R., Jacobs, P., and Skuce, A. (2013). Quantifying the consensus on anthropogenic global warming in the scientific literature. *Environ. Res. Lett*, 8:24024–7.

[35] Cortes, C. and Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20(3):273–297.

[36] Craig, P. P., Gadgil, A., and Koomey, J. G. (2002). What can history teach us? A retrospective examination of long-term energy forecasts for the United States. *Annual Review of Energy and the Environment*, 27(1):83–118.

[37] Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. (2006). Online Passive-Aggressive Algorithms. *Journal of Machine Learning Research*.

[38] Dale, M. (2013). A Comparative Analysis of Energy Costs of Photovoltaic, Solar Thermal, and Wind Electricity Generation Technologies. *Applied Sciences*, 3(2):325–337.

[39] Department for Business, E. . I. S. (2010). Average prices of fuels purchased by the major uk power producers. *UK Government*.

[40] Department for Business, E. . I. S. (2019). Companies house - gov.uk. *UK Government*.

[41] Department for Business Energy & Industrial Strategy (2016). Electricity Generation Costs. (November).

[42] Department for Business Energy & Industrial Strategy (2019). Updated energy and emissions projections 2018. *The Energy White Paper*, (April).

[43] Depuru, S. S. S. R., Wang, L., and Devabhaktuni, V. (2011). Smart meters for power grid: Challenges, issues, advantages and status. *Renewable and Sustainable Energy Reviews*, 15(6):2736–2742.

[44] D'haeseleer, W., Duerinck, J., Poncelet, K., Six, D., and Delarue, E. (2015). Impact of the level of temporal and operational detail in energy-system planning models. *Applied Energy*, 162:631–643.

[45] Dillon, T. S., Sestito, S., and Leung, S. (1991). Short term load forecasting using an adaptive neural network. *Electrical Power & Energy Systems*, pages 186–192.

[46] Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A., and Vapnik, V. (1997). Support vector regression machines. *Advances in Neural Information Processing Dystems*, 1:155–161.

[47] Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (1988). Least angle regression. *The Annals of Statistics*, 32(2):440–444.

[48] Fan, S., Chen, L., and Lee, W. J. (2009). Short-term load forecasting using comprehensive combination based on multimeteorological information. *IEEE Transactions on Industry Applications*, 45(4):1460–1466.

[49] Fard, A. K. and Akbari-Zadeh, M.-R. (2014). A hybrid method based on wavelet, ANN and ARIMA model for short-term load forecasting. *Journal of Experimental & Theoretical Artificial Intelligence*, 26(2):167–182.

[50] Fishbone, L. G. and Abilock, H. (1981). Markal, a linear programming model for energy systems analysis: Technical description of the bnl version. *International Journal of Energy Research*, 5(4):353–375.

[51] for Business Energy, D. and Strategy, I. (2019). Power stations in the united kingdom, may 2019. *Digest of United Kingdom Energy Statistics (DUKES)*.

[52] for Energy Regulation (CER), C. (2012).

[53] Forgy, E. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classification. *Biometrics*, 21(3):768–769.

[54] Forshaw, M., Thomas, N., and McGough, A. S. (2016). The case for energy-aware simulation and modelling of internet of things (IoT). *Proceedings of the 2nd International Workshop on Energy-Aware Simulation - ENERGY-SIM '16*, pages 1–4.

[55] Fortin, F.-A., De Rainville, F.-M. D. R., Gardner, M.-A., Parizeau, M., and Gagne, C. (2012). DEAP: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research*, 13:1–5.

[56] Freund, Y. and Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

[57] Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1):1–22.

[58] Friedman, J. H. (316). Greedy Function Approximation: A Gradient Boosting Machine. page 400.

[59] Gargiulo, M. and Brian, O. (2013). Long-term energy models : Principles , characteristics , focus ,. *WIREs Energy and Environment*, 2(April).

[Ghofrani et al.] Ghofrani, M., Hassanzadeh, M., and Fadali, M. S. Smart Meter Based Short-Term Load Forecasting for Residential Customers. pages 13–17.

[61] Ghorbani, A., Dechesne, F., Dignum, V., and Jonker, C. (2014). Enhancing ABM into an Inevitable Tool for Policy Analysis. *Journal on Policy and Complex Systems*, 1(1):61–76.

[62] Gillespie, J. V. (1979). Forecasting: An appraisal for policy-makers and planners. by william ascher. (baltimore: Johns hopkins university press, 1978. pp. xiii 239. $15.00, paper.). *American Political Science Review*, 73(2):554—-555.

[63] Goncalves Da Silva, P., Ilic, D., and Karnouskos, S. (2014). The Impact of Smart Grid Prosumer Grouping on Forecasting Accuracy and Its Benefits for Local Electricity Market Trading. *IEEE Transactions on Smart Grid*, 5(1):402–410.

[64] Gross, G. and Galiana, F. (1987). Short-term load forecasting. *Proceedings of the IEEE*, 75(12):1558–1573.

[65] Gross, R. (2007). Investment in electricity generation: the role of costs, incentives and risks. (May).

[66] Group, N. P. (2019). N2ex day ahead auction prices. *Nordpoolgroup.com*.

[67] Grozev, G., Batten, D., Anderson, M., Lewis, G., Mo, J., and Katzfey, J. (2005). NEMSIM: agent-based simulator for Australia's national electricity market. *SimTecT 2005 Conference Proceedings, Sydney, Australia*.

[68] Haben, S., Grindrod, P., Rowe, M., Holderbaum, W., Singleton, C., and Potter, B. (2013). Mathematical Solutions for Electricity Networks in a Low Carbon Future. *22nd International Conference on Electricity Distribution*, (0857):10–13.

[69] Hadar, A. and Hartmann, B. (2019). Simulation of cross-border capacity allocation. *1st IEEE Student Conference on Electric Machines and Systems, SCEMS 2018*, pages 1–6.

[70] Hall, L. M. H. and Buckley, A. R. (2016). A review of energy systems models in the UK : Prevalent usage and categorisation. *Applied Energy*, 169:607–628.

[71] Harp, B. S. A., Wollenberg, B. F., and Samad, T. (2000). SEPIA: A Simulator for Electric Power Industry Agents. (August):53–69.

[72] Haydt, G., Leal, V., Pina, A., and Silva, C. A. (2011). The relevance of the energy resource dynamics in the mid/long-term energy planning models. *Renewable Energy*, 36(11):3068–3074.

[73] Haydt, G., Leal, V., Pina, A., Silva, C. A., Loulou, R., Labriet, M., Taylor, J. W., Kou, Y., Lu, C.-t. T., Sirwongwattana, S., Huang, Y. P., Sinvongwattana, S., Yoh-Han, P., Dejan J., S., and Lomet, D. B. (2008). An evaluation of methods for very short-term load forecasting using minute-by-minute British data. *International Journal of Forecasting*, 24(4):645–658.

[74] Heaps, C. (2016). Long-range energy alternatives planning (leap) system.

[75] Hilbers, A. P., Brayshaw, D. J., and Gandy, A. (2019). Importance subsampling: improving power system planning under climate-based uncertainty. *Applied Energy*, 251(May):113114.

[76] Hill, T., Marquez, L., O'Connor, M., and Remus, W. (1994). Artificial neural network models for forecasting and decision making. *International Journal of Forecasting*, 10(1):5–15.

[77] Hines, P., Apt, J., and Talukdar, S. (2008). Trends in the history of large blackouts in the United States. *IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES*, 15213:1–8.

[78] Hinton, G. E. (1989). Connectionist learning procedures. *Artificial Intelligence*, 40(1-3):185–234.

[79] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

[80] Hodges, J. S. and Dewar, J. A. (1992). Is it you or your model talking? A framework for Model Validation.

[81] Hoffman, K. (2011). Perspectives on the Validation of Energy System Models Motivation. *MITRE*, (February 2006):1–13.

[82] Hoffman, K. C. (1973). The u.s. energy system: A unified planning framework, energy modeling, resources for the future. pages 103—-43.

[83] Holt, C. C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5–10.

[84] Hong, T., Wilson, J., Xie, J., and Member, A. (2014). Long Term Probabilistic Load Forecasting and Normalization With Hourly Information. 5(1):456–462.

[85] Howells, M., Rogner, H., Strachan, N., Heaps, C., Huntington, H., Kypreos, S., Hughes, A., Silveira, S., DeCarolis, J., Bazillian, M., and Roehrl, A. (2011). OSeMOSYS: The Open Source Energy Modeling System. *Energy Policy*, 39(10):5850–5870.

[86] Huang, S.-j., Member, S., and Shih, K.-r. (2003). Short-Term Load Forecasting Via ARMA Model Identification Including Non-Gaussian. 18(2):673–679.

[87] Humeau, S., Wijaya, T. K., Vasirani, M., and Aberer, K. (2013). Electricity load forecasting for residential customers: Exploiting aggregation and correlation between households. *2013 Sustainable Internet and ICT for Sustainability, SustainIT 2013*.

[88] Hunt, K., Blekicki, A., and Callery, R. (2015). Availability of utility-scale photovoltaic power plants. *2015 IEEE 42nd Photovoltaic Specialist Conference, PVSC 2015*, pages 0–2.

[89] Hunter, K., Sreepathi, S., and DeCarolis, J. F. (2013). Modeling for insight using Tools for Energy Model Optimization and Analysis (Temoa). *Energy Economics*, 40(November 2018):339–349.

[90] IEA (2015). Projected Costs of Generating Electricity. page 215.

[91] Ilić, D., da Silva, P. G., Karnouskos, S., and Jacobi, M. (2013). Impact assessment of smart meter grouping on the accuracy of forecasting algorithms. *Proceedings of the 28th Annual ACM Symposium on Applied Computing - SAC '13*, page 673.

[92] IRENA (2014). *Renewable Power Generation Costs in 2014*.

[93] IRENA (2018). *Renewable Power Generation Costs in 2017. IRENA - International Renewable Energy Agency*.

[94] Ito, K., Ida, T., and Tanaka, M. (2013). Using Dynamic Electricity Pricing to Address Energy Crises Evidence from Randomized Field Experiments. page 41.

[95] Jager, W. (2006). Simulating consumer behaviour: a perspective. *Environmental Policy and Modelling in Evolutionary Economics*, pages 1–28.

[96] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8):651–666.

[97] Johansson, C., Bergkvist, M., Geysen, D., Somer, O. D., Lavesson, N., and Vanhoudt, D. (2017). Operational Demand Forecasting in District Heating Systems Using Ensembles of Online Machine Learning Algorithms. *Energy Procedia*, 116:208–216.

[98] Jr., J. H. W. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.

[99] Kamgarpour, M. and Tembine, H. (2013). A Bayesian Mean Field Game Approach to Supply Demand Analysis of the Smart Grid. *First International Black Sea Conference on Communications and Networking (BlackSeaCom) A*, pages 211–215.

[100] Keles, D., Jochem, P., Mckenna, R., Ruppert, M., and Fichtner, W. (2017). Meeting the Modeling Needs of Future Energy Systems. *Energy Technology*, pages 1007–1025.

[101] Kell, A., Forshaw, M., and Mcgough, A. S. (2019). ElecSim : Monte-Carlo Open-Source Agent-Based Model to Inform Policy for Long-Term Electricity Planning. *The Tenth ACM International Conference on Future Energy Systems (ACM e-Energy)*, pages 556–565.

[102] Kell, A., McGough, A., and Forshaw, M. (2018a). Segmenting residential smart meter data for short-Term load forecasting. In *e-Energy 2018 - Proceedings of the 9th ACM International Conference on Future Energy Systems*.

[103] Kell, A., Mcgough, A. S., and Forshaw, M. (2018b). Segmenting Residential Smart Meter Data for Short-Term Load Forecasting. *e-Energy Conference*, pages 91–96.

[104] Kell, A. J. M., Forshaw, M., and McGough, A. S. (2020). Long-Term Electricity Market Agent Based Model Validation using Genetic Algorithm based Optimization. *The Eleventh ACM International Conference on Future Energy Systems (e-Energy'20)*.

[105] Kim, K.-h., Youn, H.-s., Member, S., and Kang, Y.-c. (2000). Short-term load forecasting for special days in anomalous load conditions using neural networks. *IEEE Transactions on Power Systems*, 15(2):559–565.

[106] Kincheloe, S. C. (1990). The weighted average cost of capital - the correct discount. *The Appraisal journal.*, 58(1).

[107] Koomey, J., Craig, P., Gadgil, A., and Lorenzetti, D. (2003). Improving Long-Range Energy Modeling: A Plea for Historical Retrospectives. *Energy Journal*, 24(4):75–92.

[108] KPMG (2017). Cost of Capital Study 2017. *KPMG*.

[109] Künzel, T., Gmbh, F., Kg, C., Klumpp, F., Gmbh, F., and Kg, C. (2018). Bidding Strategies for Flexible and Inflexible Generation in a Power Market Simulation Model. pages 532–537.

[110] Law, A. M. and Kelton, D. W. (2000). *Simulation modeling and analysis; 3rd ed.* McGraw Hill Series in Industrial Engineering and Management Science. McGraw-Hill, New York, NY.

[111] Li, Y., Guo, P., and Li, X. (2016). Short-Term Load Forecasting Based on the Analysis of User Electricity Behavior. *Algorithms*, 9(4):1–14.

[112] Liu, X., Golab, L., Golab, W., Ilyas, I. F., and Jin, S. (2016). Smart Meter Data Analytics: Systems, Algorithms, and Benchmarking. *ACM Transactions on Database Systems*, 42(1):1–39.

[113] Ltd, L. (2016). Final Report: Electricity Generation Costs and Hurdle Rates.

[114] Lu, C., Wu, H. T., and Vemuri, S. (1993). Neural network based short term load forecasting. *IEEE Transactions on Power Systems*, 8(1):336–342.

[115] Ludig, S., Haller, M., Schmid, E., and Bauer, N. (2011). Fluctuating renewables in a long-term climate change mitigation strategy. *Energy*, 36(11):6674–6685.

[116] Ma, J., Saul, L. K., Savage, S., and Voelker, G. M. (2009). Identifying suspicious URLs: An application of large-scale online learning. *Proceedings of the 26th International Conference On Machine Learning, ICML 2009*, pages 681–688.

[117] Machado, P. G., Mouette, D., Villanueva, L. D., Esparta, A. R., Mendes Leite, B., and Moutinho dos Santos, E. (2019). Energy systems modeling: Trends in research publication. *Wiley Interdisciplinary Reviews: Energy and Environment*, 8(4):1–15.

[118] Mahmood, A., Ullah, M. N., Razzaq, S., Basit, A., Mustafa, U., Naeem, M., and Javaid, N. (2014). A new scheme for demand side management in future smart grid networks. *Procedia Computer Science*, 32:477–484.

[119] Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate analysis*. Academic Press London ; New York.

[120] Masson-Delmotte, V., Zhai, P., Pörtner, H., Roberts, D., Skea, J., Shukla, P. R., Pirani, A., Moufouma-Okia, W., Péan, C., Pidcock, R., Connors, S., Matthews, J. B., Chen, Y., Zhou, X., Gomis, M. I., Lonnoy, E., Maycock, T., Tignor, M., and Waterfield, T. (2018). *IPCC Special Report 1.5 - Summary for Policymakers*. IPCC.

[121] Möst, D. and Keles, D. (2010). A survey of stochastic modelling approaches for liberalised electricity markets. *European Journal of Operational Research*, 207(2):543–556.

[122] Nagi, J., Yap, S. K., Tiong, S. K., and Ahmed, S. K. (2008). Electrical Power Load Forecasting using Hybrid Self-Organizing Maps and Support Vector Machines. *The 2nd International Power Engineering optimization Conference (PEOCO)*, (June):51 – 56.

[123] Nahmmacher, P., Schmid, E., Hirth, L., and Knopf, B. (2016). Carpe diem: A novel approach to select representative days for long-term power system modeling. *Energy*, 112:430–442.

[124] National Grid (2019). STOR Market Information Report. (October):0–11.

[125] Nguyen, H. and Hansen, C. K. (2017). Short-term electricity load forecasting with Time Series Analysis. *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pages 214–221.

[126] Noble, K. (2004). ANSWERv6-MARKAL User Manual ANSWER MARKAL , an Energy Policy Optimization Tool. *Energy Policy*.

[127] of Energy, D. and Change, C. (2013). Energy act 2013. *UK Government*.

[128] Pao, H.-T. (2007). Forecasting electricity market pricing using artificial neural networks. *Energy Conversion and Management*, 48(3):907–912.

[129] Paper, I. W. and Heptonstall, P. (2012). Cost estimates for nuclear power in the UK. *ICEPT Working Paper*, (August).

[130] Perloff, J. M. (2012). *Microeconomics Sixth Edition*.

[131] Pfenninger, S. and Staffell, I. (2016). Long-term patterns of European PV output using 30 years of validated hourly reanalysis and satellite data. *Energy*, 114:1251–1265.

[132] Poncelet, K., Hoschle, H., Delarue, E., Virag, A., and Drhaeseleer, W. (2017). Selecting representative days for capturing the implications of integrating intermittent renewables in generation expansion planning problems. *IEEE Transactions on Power Systems*, 32(3):1936–1948.

[133] Poonpun, P. and Jewell, W. (2008). Analysis of the cost per kWh to store electricity. *IEEE Transactions on Energy Conversion*, 23(2):529–534.

[134] Praça, I., Ramos, C., Vale, Z., and Ascem, M. (2003). MASCEM : A Multiagent Markets.

[135] Quilumba, F. L., Lee, W.-j., Huang, H., Wang, D. Y., Member, S., and Szabados, R. L. (2014). Using Smart Meter Data to Improve the Accuracy of Intraday Load Forecasting Considering Customer Behavior Similarities. pages 1–8.

[136] Ringler, P. (2012). PowerACE Agent-based simulation of electricity markets. *MOCAP Workshop on Modelling Carbon Prices - Interacting agent networks & Strategies under risk*.

[137] Ringler, P., Keles, D., and Fichtner, W. (2016). Agent-based modelling and simulation of smart electricity grids and markets - A literature review. *Renewable and Sustainable Energy Reviews*, 57(September):205–215.

[138] Roth AE and Erev I (1995). Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. *Games and economic behavior*, 8(1):164–212.

[139] Rothengatter, W. (2007). Assessment of the impact of renewable electricity generation on the German electricity sector An agent-based simulation approach.

[140] Sandbag (2020). Carbon price viewer - sandbag. *sandbag.org.uk/carbon-price-viewer/*.

[141] Saxena, K. and Abhyankar, A. R. (2019). Agent based bilateral transactive market for emerging distribution system considering imbalances. *Sustainable Energy, Grids and Networks*, 18:100203.

[142] Schmitt, J., Hollick, M., Roos, C., and Steinmetz, R. (2008). Adapting the user context in realtime: Tailoring online machine learning algorithms to ambient computing. *Mobile Networks and Applications*, 13(6):583–598.

[143] Schrattenholzer, L. (1981). The energy supply model MESSAGE. *European Journal of Operational Research*, (December).

[144] Shu, F. and Luonan, C. (2006). Short-term load forecasting based on an adaptive hybrid method. *Power Systems, IEEE Transactions on*, 21(1):392–401.

[145] Singh, A. K., Ibraheem, Khatoon, S., Muazzam, M., and Chaturvedi, D. K. (2012). Load forecasting techniques and methodologies: A review. *ICPCES 2012 - 2012 2nd International Conference on Power, Control and Embedded Systems*.

[146] Sioshansi, F. P. (2009). *Competitive electricity markets: design, implementation, performance*. Elsevier.

[147] Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222.

[148] Staffell, I. and Pfenninger, S. (2016). Using bias-corrected reanalysis to simulate current and future wind power output. *Energy*, 114:1224–1239.

[149] Sun, J. and Tesfatsion, L. (2007). Dynamic Testing of Wholesale Power Market Designs : An Open-Source Agent-Based Framework. *Computational Economics*, 30(3):291–327.

[150] Suna, D. and Resch, G. (2016). Is nuclear economical in comparison to renewables? *Energy Policy*, 98:199–209.

[151] Tang, L., Wu, J., Yu, L., and Bao, Q. (2015). Carbon emissions trading scheme exploration in China: A multi-agent-based model. *Energy Policy*, 81(2015):152–169.

[152] Telefónica (2014). The Smart Meter Revolution. Towards a smarter future. page 11.

[153] Theodoridis, S. (2009). *Pattern recognition*. Academic Press, Burlington, MA London.

[154] Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

[Tin Kam Ho] Tin Kam Ho. Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 1:278–282.

[156] Usher, A. P. (1961). Energy in the american economy, 1850–1975. an economic study of its history and prospects. *The Journal of Economic History*, 21(03):418–421.

[157] V. Braitenberg, G. Palm, D. G. P. D. A. A. (1986). *Brain Theory: Proceedings of the First Trieste Meeting on Brain Theory, October 1–4, 1984*. Springer-Verlag Berlin Heidelberg, 1 edition.

[158] Viegas, J. L., Vieira, S. M., Melício, R., Mendes, V. M., and Sousa, J. M. (2016). Classification of new electricity customers based on surveys and smart metering data. *Energy*, 107:804–817.

[159] Vrablecová, P., Bou Ezzeddine, A., Rozinajová, V., Šárik, S., and Sangaiah, A. K. (2017). Smart grid load forecasting using online support vector regression. *Computers & Electrical Engineering*, 0:1–16.

[160] Weidlich, A. and Veit, D. (2008). A critical survey of agent-based wholesale electricity market models. *Energy Economics*, 30(4):1728–1759.

[161] White, A. (2005). Concentrated power. *Public utilities fortnightly. Arlington*, 143(2):43–47.

[162] Wiener, N. (1930). Autoregressive integrated moving average.

[163] Wijaya, T. K., Vasirani, M., Humeau, S., and Aberer, K. (2010). Residential Electricity Load Forecasting : Evaluation of Individual and Aggregate Forecasts. pages 1–8.

[164] Witten, I. H., Frank, E., and Hall, M. a. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*.

[165] Zhou, Z., Chan, W. K., and Chow, J. H. (2007). Agent-based simulation of electricity markets: A survey of tools. *Artificial Intelligence Review*, 28(4):305–342.