ISSN/0393-3016

MARCO RAO
Department of Fusion and Technology for Nuclear Safety and Security
Division of Technology Applications for Security, Health and Heritage
Development of Particle Accelerators and Medical Applications Laboratory
Frascati Research Centre, Rome

GIADA DE ANGELIS
University of Siena

ELENA SEBASTIANELLI
University of Rome "La Sapienza"

TEACHING A MACHINE TO THINK AT ENERGY COST

RT/2022/4/ENEA

ITALIAN NATIONAL AGENCY FOR NEW TECHNOLOGIES,
ENERGY AND SUSTAINABLE ECONOMIC DEVELOPMENT

MARCO RAO
Department of Fusion and Technology for Nuclear Safety and Security
Division of Technology Applications for Security, Health and Heritage
Development of Particle Accelerators and Medical Applications Laboratory
Frascati Research Centre, Rome

GIADA DE ANGELIS
University of Siena

ELENA SEBASTIANELLI
University of Rome "La Sapienza"

TEACHING A MACHINE TO THINK AT ENERGY COST

RT/2022/4/ENEA

ITALIAN NATIONAL AGENCY FOR NEW TECHNOLOGIES,
ENERGY AND SUSTAINABLE ECONOMIC DEVELOPMENT

TEACHING A MACHINE TO THINK AT ENERGY COST

Marco Rao, Giada De Angelis, Elena Sebastianelli

Abstract

This work concerns the application of some machine learning (ML) methods to the problem of estimating the levelised cost of energy (LCOE) by class of electricity generation technologies: in the case study, a photovoltaic plant of 0.83 MW is considered using national data provided by GSE for the International Energy Agency in 2020. The purpose of this work is to highlight the pros and cons of the application of ML in the improving of accuracy in evaluation of such a cost compared to more traditional methods such as linear regression models, and to propose ideas for further insights on their use in this context.

Key words: *Machine Learning, LCOE.*

Riassunto

Questo lavoro concerne l'applicazione di alcune metodi di machine learning (ML) al problema della stima del costo livellato dell'energia (LCOE) per classi di tecnologie di produzione energetica: nel caso studio, si considera un impianto fotovoltaico di 0.83 MW di potenza come censito da GSE a livello nazionale per l'Agenzia Internazionale dell'Energia nel 2020. Scopo di questo lavoro, evidenziare pro e contro dell'applicazione del ML per l'incremento di accuratezza nella valutazione dei costi medesimi rispetto a metodi più tradizionali come i modelli di regressione lineare, e proporre spunti per approfondimenti ulteriori sul loro impiego in tale contesto.

Parole chiave: Machine Learning, LCOE.

Summary

Introduction

This work proposes some applications of several machine learning[1] (ML) approaches in energy system analysis: in particular, the proposed work deal with the energy cost, using the theoretical framework provided by the IEA methodology [**3**] about the so-called Levelised Cost of Energy (LCOE).

The authors are aware that the proposed work only investigates a small part of the applications of ML to the energy system: nonetheless, this exercise, albeit limited, is highly significant of the potential of ML in this field, which are certainly not limited, nor are they intended, as devoted to improves performance in the execution of specific tasks.

The work is structured as follows.
The two methodological cornerstones of the work are discussed in detail in the follows.
A discussion of LCOE and on the fundamentals of Machine Learning is provided to understand the performed work, which consists in the development of predictive models to estimate the LCOE of the selected technology given some explanatory variables.
The explanatory variables are the ones reported in the LCOE standard model as provided by the IEA.
The performed simulation aimed to compare the used ML techniques with a canonical method, namely a traditional linear model, reporting the differences in terms of two well-known standard accuracy measures, such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

Fig. 1 and 2

[1] There are many excellent starting points to enter in the ML world: here we just recall the ones provided by TensorFlow platform from Google [**1**] and by the excellent text of Chollet [**2**].

1 The levelised cost of energy (LCOE)

1.1 The basis

One of the most common, worldwide accepted, measures to compare alternative sources of energy is the so-called *levelized cost of energy*, or levelized cost of electricity (LCOE) provided by IEA [**3**][2]. LCOE is, literally, *"the average revenue per produced electricity unit necessary to recover initial investments and operating costs over the expected financial and operational power plant's life cycle"* [**4**].

The LCOE is the average cost, express in some currency (usually, the U.S. dollar) per energy unit, and is equal to the Total Life Cycle Cost of a certain technology by the discounted energy yields obtained from the production in a defined time horizon. The LCOE allowed cost ranking in comparing technologies also radically different technically and it can be also used in time series analysis or for determination of feed-in-tariffs and grid parity analyses [**5**].

The LCOE was chosen to perform the proposed analysis cause, in a macroeconomic perspective, it is one of the most suitable one and its formula can easily be adapted to many types of investigations required in decision support analysis in policy making.

In the follows, a detail of the LCOE formula building is proposed, to better understand the meaning of this indicator.

1.1.1. From UCOE to LCOE

The logic behind the LCOE formula is something commonly used in Economics: it can be seen as a *cost-benefit* ratio, considered cash flows discounted at the t when the technology is analyzed, given the overall characteristics of the technology itself.

In the following formulas, C_t, is the total capital construction costs in year t, $O\&M_t$, operation and maintenance costs in year t, and F_t, fuel costs in year t, $O\&M_t$ is the operation

[2] See, especially for the methodological part, section II of 2020 report.

and maintenance costs in year *t*, and F_t, the fuel costs in year *t*, and E_t, the amount of energy produced in year *t*.

With such parameters we can get the *undiscounted cost of energy (UCOE)*:

Eq.1

$$UCOE = \frac{\sum_{t=1}^{n} C_t + O\&M_t + F_t}{\sum_{t=1}^{n} E_t}$$

There's no need to explain that only introducing a *discount factor* we can capture time uncertainty and comparing costs and production related to different periods became possible. A discount factor is generally defined as in *Eq.2*.

Eq.2

$$Discount\ Factor = \frac{1}{(1+d)^t}$$

The discount factor formula considers the time horizon *t,* namely the expected lifetime of the plant, and the discount rate *d*, calculated by the analyst. A discount rate could either be nominal, when it includes the effects of inflation, or real, when those effects are excluded: in this study we follow the IEA suggestions, so the real discount rate will be taken in consideration, known that this is a necessary simplification for the analyst about the role of inflation in real life on several factors, like commodities prices.

The discount factor can be first applied on the flow of costs sustained during the period, obtaining the *total life-cycle cost* (TLCC):

Eq.3

$$TLCC = \sum_{t=1}^{n} \frac{C_t + O\&M_t + F_t}{(1+d)^t}$$

In TLCC the cost flow is discounted to a base year: it is a useful measure for comparing mutual exclusive projects, however there is no frame reference to decide if the value of TLCC indicates a beneficial investment, so it's not recommended for economic evaluation [**6**]. The TLCC is conceptual improvements respect the UCOE but misses the

benefits, since the amount of energy produced during the period is not considered. Dividing the TLCC by the summatory of energy produced in the period *t*, it is possible to obtain the *discounted costs cost of energy* (DCCOE), as in *Eq.4*.

Eq.4

$$DCCOE = \sum_{t=1}^{n} \frac{C_t + O\&M_t + F_t}{(1+d)^t} / \sum_{t=1}^{n} E_t$$

The DCCOE consider the energy yields but only the costs are discounted: in practice, is only one step away from the simplest version of the LCOE formula. In fact, the DCCOE differs from the classical form of the LCOE only eliminating the discount factor in the second part of the equation[3]: since energy generation in all technologies is almost constant every year, the ratio of LCOE to DCCOE is expected to be almost constant as well for each technology considered [**7**].

Now we're ready to consider the standard LCOE formula by the IEA:

Eq.5

$$LCOE = \sum_{t=1}^{n} \frac{C_t + O\&M_t + F_t + D_t + CO2_t}{(1+d)^t} / \sum_{t=1}^{n} \frac{E_t}{(1+d)^t}$$

In the LCOE we divide the discounted sum of costs by the discounted sum of energy production on a lifetime for the considered technology: it's a cost measure that includes all the others and appears as a simplest, yet most refined and easiest to interpret, cost of energy that could be found in literature [**8**].

In the IEA formulation, two additional costs are considered: D_t, the decommissioning cost in year *t*, and $CO2_t$, the cost of carbon emissions in year *t*. Adding those two elements is essential for sustainable decisions in energy production: D_t accounts for the costs that have to be sustained at the end of the power plant's lifetime, (it's a sort of costs for *"waste management"* in a circular economy perspective); CO2 cost accounts for the carbon impact of chosen technology highlighting the significant difference existent between fossil fuels-based plants and the renewable ones.

[3] This is true when decommissioning and Co2 costs are not included.

1.2 Some improvements of the standard LCOE formula

The LCOE formula is deeply simplified, and several authors tried to rediscuss the LCOE, to address specific issues in energy analysis: in the follows we discuss some of these improvements and proposal.

1.2.1 The NREL formula

A good alternative proposal to the IEA basic formulation it is the one presented by the *National Renewable Energy Laboratory* (NREL): it defines the LCOE in terms of annual cost of energy and bases the calculation on the costs of financing the project capital. The NREL formulation includes the *capital recovery factor* (CRF), defining the LCOE as the minimum price at which energy must be sold for an energy project to break even [**9**].

Eq.7

$$LCOE_{NREL} = \frac{CRF \times C_o + O}{8760 \times CF} + F * h + V$$

C_o is the overnight cost of capital ($/kW), O is the fixed operating and maintenance costs ($/kW-yr), *CF is* the capacity factor multiplied by 8760, the number of hours in a year, *F* the fuel cost, expressed in dollars per million British thermal units ($/MMBtu) multiplied by *h*, heat rate, measured in British thermal units per kilowatt-hour (Btu/kWh), *V* is the variable O&M costs ($/kWh). The LCOE by NREL don't create interpretation problems even if the formula it is taken out of context since each component is directly expressed in monetary units[4]. The logic behind the formula is the same that the IEA one: the total cost over an annual period is calculated and divided by the energy produced in the same period.

The time factor *t* and the interested rate *i* are considered through the *Capital Recovery Factor* (CRF) formula, expressed in *Eq.8*

Eq.8

$$CRF = \frac{i(1+i)^n}{(1+i)^n - 1}$$

[4] See, at this regard, the annotation of IEA at page 35 of [3].

The discount rate *i* is the main component of the formula, together with *n*, the number of annuities received, or, in a more intuitive way, the number of payments made to repay the capital (Aldersey-Williams and Rupert, 2019). The CRF is the main financial assumption in this formulation: the NREL defines it has *"the ratio of a constant annuity to the present value of receiving that annuity for a given length of time."*, in this way the present value is expressed in terms of the annuity, the interest rate, and the number of annuities. Eventual doubts regarding which discount rate should be used, according to expectations and uncertainty, are the same in the IEA formula. There are no substantial advantages in using one formula rather than another, except for the focus that the analyst wants to give to its work[5].

The IEA formula and the NREL formula return the same value of LCOE under simplified assumptions [**8**]: not considering decommissioning costs and CO2 in the IEA formula; assuming a constant annual output for the project as well as for costs; if the initial construction fixed costs must be all in the year 1 and capital recovery starts immediately with a financing term equal to the project's operating life. If those assumptions are true, the two formulas are perfectly equivalent, and one is perfectly replaceable with the other.

1.2.2 WACC as an alternative to fixed discount rate

Using a fixed value of the discount rate for every technology, without taking in consideration the difference sources of funding used for each single project is controversial. A more proper measure to calculate the discount rate could be the *Weighted Average Cost of Capital* (WACC) that provides an average measure of the cost of capital weighting each type of financing using the ratio between every single type of source and the total amount of capital. Capital is usually composed by debt, and common equity, the amount of capital invested by the shareholders, which is reported in the WACC formula, in Eq.9.

Eq.9

$$WACC = C_e * \frac{E}{E+D} + C_d * (1-T) * \frac{D}{D+E}$$

[5] The NREL formula gives more attention to investment and capital recovery, so it could me more suitable for a private investor, while the IEA formula, including social costs, focuses more on public side of the project and can be easily used by the decision maker.

C_e is the cost of the common equity, C_d is the cost of debt, E is the weight of common equity on the total liability, D is the weight of debt on the total liability, $E+D$ the total liability and T the corporate tax rate. Both the cost of common equity and the cost of debt are expressed in term of rate of return of the single financial source, where the rate of return is a measure used to establish the profitability of an investment[6].

There are two main reasons why WACC should be used in substitution of the standard discount rate for every technology in the LCOE formula: incorporating the risk regarding the financing of the project and taking in consideration the fiscal regime. Using the same fixed discount rate for every technology does not consider the risk deriving from using different sources for financing therefore the risk associated with the single project. The WACC express the cost of investing in a certain project, therefore the investors want to receive back from the project at least this return or either the project is not worth to invest in. Using the WACC as a discount rate rises the LCOE for those technologies considered financially riskier and potentially advantages those ones that are apparently less risky [**8**]: they also reports how investors in the energy sector match their expectations regarding return with the available technologies and the market trends: financial investors whit lower return expectations, such as pension funds and insurance companies, are investing in offshore wind projects and this trend is indicative that the perception is that the risk related to the project will decrease as the technology matures.

Secondly, as already discussed, the standard LCOE formula does not consider the fiscal regime in which the project is inserted. However, fiscal policy instruments are one of the most applied strategies to influence economic players and this is also true in green energy production. This means that different technologies can have a different tax rate according to the national pollution standards, for example. In the WACC formula, the corporate tax rate T is multiplied for the cost of debt: for the single project, the cost of debt is also weighted for the level of taxation applied to the power plant considered. However, this can be considered just a partial solution for including fiscal regime in the LCOE formula: pollution taxes, for example, are applied not as a corporate tax but as a tax per unit of pollution produced, considering a certain pollutant[7].

[6] The rate of return differs whether the utility is regulated or not: in case of a regulated utility, such electricity, the suggested value is 9.1% for before-nominal cost of debt and 13.0% for nominal cost of equity.

[7] CO2 taxes per unit emitted are the most common one, but taxes can be applied on every other pollutant substance, for example on a certain quantity of pollutant released in water, a river for example.

1.2.3 Power purchase agreements in LCOE formula

Another interesting point is related to the several typologies of incentives that RES power plants have access to in fact, a popular instrument used to guarantee a fair and risk-controlled agreement in selling and buying RES produced electricity is using the *Power Purchase Agreements* (PPAs). As already discussed, nowadays the biggest concern regarding RES power production is related to their technological unfeasibility in producing a time and amount-controlled stock of electricity, which creates a competitive disadvantage in the electricity market against old school technologies. PPAs are a type of contract in which a seller, the electricity producer, and a buyer, the subject willing to purchase the electricity produced, agrees in terms of quantity of electricity produced.

The idea behind a common PPA contract is that a private subject is willing to use green power for its matters but has no intention in investing its own capital to install a renewable technology: this subject can stipulate a PPA contract with a seller which owns these technologies and its willing to sell in a more legally protected environment. The seller can choose even to provide the installation of the technology while the only duty of the buyers is buying power on a KWh basis. A slightly different version of a common PPA is a *virtual* PPA, where the seller chooses to compete on the market selling power to the grid and signs an agreement whit a third-party investor which guarantees the owner a certain fixed price for the electricity sold to the grid. The advantage for the third-party is that the if the grid price for electricity is higher than the agreed one, the difference results as a profit for the investor. PPAs are widely diffused in big renewable electricity producers like Europe, the U.S and Latin America.

Brucks et al [**10**] show there is an interesting correlation between PPAs and the calculation of LCOE for wind farms production: theoretically, PPAs use an LCOE model to determine which should be the fair price for the energy produced, considering the whole cost profile over the expected. However, PPAs has the power to contribute to the cost profile since the aim itself of a PPA contract is to share and reduce the risk of additional costs, which are not accounted in the LCOE model.

This allocation of risk is negotiated between the subject involved in the contract and reflects the market condition of the specific country in which the contract is signed, and the technology involved, which is really like what happen in the calculation of the LCOE. It is

reasonable to hypothesize that a PPA contract influences the LCOE of a certain technology where the contract is applied.

In their study, the real data from 7 windfarms located in Germany and Denmark were used to study the effects of PPAs on LCOE. The authors wanted to include energy delivery limits and their penalty cost in a newly improved LCOE model: energy delivery limits can be in form of maximum annual energy delivery limit, minimum one, both together and neither. Since the energy delivery limits have a direct correlation with an already existing component of the LCOE formula, the first step is defining the cost of underdelivering energy PN_t as the difference between the energy generated and the one delivered, considering the expected energy production P_{exp} that the buyer requires, as follows in *Eq.10*.

Eq.10

$$PN_t = \begin{cases} (Min_{lim}P_{exp} - E_t)COE_t & when\ E_t < Min_{lim}P_{exp} \\ 0 & when\ E_t \geq Min_{lim}P_{exp} \end{cases}$$

COE_t is the agreed cost of energy between the seller and the buyer, E_t is the amount of energy generated in year *t*, which is already a component of the LCOE formula, Min_{lim} is the smallest fraction of expected energy production P_{exp} that is required by the buyer. What happens is that in the case in which the seller does not meet the production requirements is obliged to compensate the buyer in the case where it is forced to buy electricity on the spot market for a higher price than the agreed one. There is also a production loss PL_t, defined as the difference between the energy generated in year *t* and the threshold for the maximum penalty Max_{lim}, considering again the expected energy production P_{exp}, defined in *Eq.11*.

Eq.11

$$PL_t = \begin{cases} (E_t - Max_{lim}P_{exp})COE_t(1 - PPA_{term}) & when\ E_t > Max_{lim}P_{exp} \\ 0 & when\ E_t \leq Max_{lim}P_{exp} \end{cases}$$

In this case, the penalty is applied in case the production exceeds the expected agreed: the PPA_{term} establish the type of penalty applied according to the contract, for example if the PPA has a no outside sell option, this term equals to zero, while if all the electricity produced is purchase by the buyer, therefore the term is 1, the penalty is not applied. In the LCOE formula, the component Pen_t, defined as the sum of both the two penalties type is added in

the cost profile, so in the nominator of the fraction together with the types of tax credit TC_t applied to the single wind farm, as shown in *Eq.12*.

Eq.12

$$LCOE = \sum_{t=1}^{n} \frac{C_t + O\&M_t + F_t + Pen_t - TC_t}{(1+d)^t} \Big/ \sum_{t=1}^{n} \frac{E_t}{(1+d)^t}$$

The results of the study show that, using the same Max_{lim} and Min_{lim}, introducing a maximum penalty has no significant effect in the LCOE calculation, while introducing minimum penalties or both results in higher LCOEs than the ones calculated without considering the PPAs applied. Since PPAs use an LCOE model to establish the price for electricity sold in the contract, it is clear how including the PPA penalties in the LCOE calculation is fundamental for setting fair conditions for both the parts.

Most of all, calculating the LCOE with PPA penalties is a key asset in both public and private investments decision since it reduces drastically the default probability of a RES technology under a PPA agreement. This formulation in interesting because together with including the effects, also integrates the fiscal regime which the power plants are operating under still using a fixed discount rate[8], but including a specific term for the fiscal credits. However, even if the authors chose to use the IEA formulation of LCOE, decommissioning costs and CO2 are not included: while comparing older technologies with RES ones, this type of costs can make the difference in evaluating a performance[9].

1.2.4 Montecarlo simulations for managing uncertainty

Another possible model improvement applies to the uncertainty related to the variables included in the model: again, this applies especially to RES technology, since they are the ones more subject in uncertainties, both in technical and economic performance. Tran et al [**10**] incorporate a sensitivity analysis in LCOE calculation to have a more significant comparison between well-developed technologies and RES ones. The fundamental

[8] The authors set the discount rate at 0 and 8.9% per year and do not justify it theoretically.
[9] The authors' original purpose is not creating a comprehensive LCOE model, including all the possible costs, but focusing on studying PPA effects on the LCOE model. However, since the aim of this research is providing a possible improved LCOE model for strategic decisions, the authors want to highlights this lack in the theorization.

assumption is that RES technologies are more affected by uncertainty in input data since they have been in use for a shorter amount of time and less widely diffused.

Furthermore, RES technologies are non-dispatchable, which has a direct effect on generation cost due to the difference between the scheduled energy produced and the market demand, as well as geographically dependent, where is certain locations the generation cost drops because of particularly prosperous conditions. Moreover, capital cost and O&M costs are affected by uncertainty for all technologies, regarding the difference between the expected performance and the actual one. The authors firstly chose a Monte Carlo simulation[10] approach for studying uncertain conditions in O&M costs, energy production, lifetime, and reliability to perform a global sensitivity analysis: the procedure consists in interpreting the statistical distribution of an input variable, generate random variable value based on the distribution, using them to calculate the LCOE, and lastly aggregated the results, obtaining a LCOE distribution. Secondly, they tried to provide a forecast of greenhouse gasses (GHG) emission costs according to different carbon pricing scenarios.

The LCOE is again defined as the ratio between the *Total Lifetime Cost* and the *Total Lifetime Energy Production*, but the formula used is the one of *Eq.7* defined by the NREL, which means that decommissioning costs while CO2 costs are added, but an equation is used to specify the cost of CO2 for each technology. Defining an equation for CO2 helps in randomizing a value of the variable an performing a Monte Carlo simulation: the cost of CO2 is defined in *Eq.12*.

Eq.12

$$CO2 = CP \times HR \times EC \times CF \times 8760 \times 10^{-9}$$

CP is the carbon price, *HR* is the heat rate, *EC* is the life cycle GHG emissions coefficient of the technology, *CF* is again the capacity factor.[11] Using the first annual CO2 cost calculated, the future annual CO2 costs can be defined as in *Eq.13*.

[10] The Monte Carlo methods are a class of computational algorithms based on repeated random sampling to get numerical values of some observed phenomena. Their use in physical and mathematical problems is massive especially in the fields of optimization, numerical integration.

[11] The metrics for those costs are the ones that follow *CP* is expressed in $/metric, *HR* in Btu/kWh, *EC* in kg/MMBtu. The life cycle GHG emissions data is obtained from NREL's life-cycle assessment meta-analysis. Lastly, the study considers the carbon prices between $5/metric ton and $30/metric ton.

[13] This is even more clear in the IEA formulation, since the energy produced E_t appears directly and not in the form of the *CF*.

Eq.13

$$FV_t = CO2 \times \frac{(1+i+k)^n - 1}{i+k}$$

The future annual payment FV_t is obtained multiplied *CO2* by a factor made by the discount rate *i* plus an annual price change *k*, assumed to be between 3% and 5%. The future payments are summed for every year *t*, obtaining the CO2 payments PV_{CO2}. Those are summed to the overnight capital cost and multiplied by the *Capital Recovery Factor* (CRF), as in *Eq.14*.

Eq.14

$$LCOE_{NREL} = \frac{CRF \times (C_o + PV_{CO2}) + O}{8760 \times CF} + F * h + V$$

The additional uncertain variables in the stochastic model are the energy performance, the capital costs, the operation and maintenance costs, lifetime, and economics. Energy performance is defined as *"the amount of energy converted from a certain type of technology"* and it is connected to the capacity factor *CF*, defined as *"the ratio of the actual output to the maximum potential output"*. A higher production of energy reduces the LCOE, mathematically it can be seen because the *CF* appears in the denominator of the fraction, logically there is a net benefit from the electricity sold which compensates for the costs sustained[12]. Capital costs varies across technology, country, system size and in the case of RES are the major component in the cost profile since O&M have a greater impact for conventional technologies. Concern the lifetime, the major problem is in the case in which the lifetime is estimated, so newer technologies have a greater uncertainty regarding this component. Lastly with the general term *"economics"*, the authors refer to chosen discount rate, which is assumed to vary between 3% and 10%.

The peculiarity of this research is that a wide range of technologies is taken in consideration: the authors perform the Montecarlo analysis for widely known technology,

[14] Wave energy is electrical energy converted from waves' kinetic energy; tidal energy harvests the energy from tides caused by gravitational pulls of the moon. The first one is under development, while the second has no commercial operating use now. It is interesting to note that the authors choose these two technologies while not including hydrogen energy production.

RES or not, like coal, gas, nuclear, solar PV, and thermal, offshore, and onshore wind, but also for less diffused technologies like wave energy and tidal energy[13]. For all the technologies involved in the study, a distribution of the data regarding the uncertainty variable was supposed, using data referring to the US energy market and production. Each simulation considered for each way of electricity production accounts for 10,000 possible combinations of the uncertain variables. The results met what was hypothesized *a priori* by the researchers: traditional technologies are less affected by uncertainty than renewable ones, especially in the first scenario where CO2 is not included. In fact, for coal-fired, natural-gas and nuclear power generates more predictable results: LCOEs for coal power plants shows a log-normal distribution, as well as the nuclear power plants, while the natural gas ones show a normal one.

The maturity of these technologies makes they variation range less wide, which makes them more predictable when computing their LCOE. Regarding RES, the interesting find is that the technologies with lower and less uncertain LCOEs are also the ones that are more popular regarding their diffusion and usage. However, the variation ranges are inevitably vaster than non-renewable powered technologies, due to operational characteristics. The distribution for solar PV, solar thermal, offshore, and inshore wind presents a distribution close to a log-normal one. It is interesting to note how the probability distribution for offshore wind has a longer right tail, indicating the low probability of high values of the LCOE: the authors report how the data available for offshore wind energy are scarce since the technology is still in its penetration phase in the market. Again, *learning by doing* is the key for drastically reducing the unitary cost for energy produced: only by improving technological knowledge is possible to reduce the initial capital costs that rise and make more uncertain the LCOE for RES technologies.

In the second Monte Carlo simulation, CO2 cost, as described, and calculated before is added: in particular, the authors accounts for life cycle GHG production and not only for the CO2 emitted during the production, which provides a more complete perspective of the impact of each technology. Clearly, traditional power plants' LCOE, like coal powered and natural gas powered[14], critically increase when the cost of emitting GHG is included. It is interesting to notice how also onshore wind and solar PV shows an increase in LCOEs

[14] Nuclear power plants are not considered since there is no GHG emission from them. A discussion about the possible advantages and disadvantages of this condition was already provided when introducing discount rate and decommissioning costs.

possible values: RES power plants are not completely carbon neutral, especially if the entire life cycle is analyzed, and solar PV results in being more pollutant than wind turbine when it comes to GHG emissions.

The study underlines how not including GHG emissions for energy production can mislead in investments choice, preferring traditional technologies to renewable ones without accounting for the externalities produced just because they result in being more financially appealing. The choice of using a life-cycle measure for CO2 it is interesting, however data of *life-cycle assessment* (LCA) for certain technology cannot always available. LCA requires a conspicuous quantity of data and specific expertise to be performed, which means additional costs and time usage when performing an analysis. Even if LCA analysis for a certain technology are already available, it is possible that those are extremely specific, and not easily generalizable[15].

What it is made clear in this research it is necessary to evaluate how to include a certain measure of the CO2, as well as those specific variables have a greater impact on the variation range of the LCOE. Capital cost seems to be still on of the components that clearly has the greater influence on the variation range of the LCOE: focusing on the behavior of a specific component, modelling its dynamic and verify how a forecast impacts on the LCEO can be useful when it comes to planning strategic decisions. The authors proposed possible solutions in reducing the cost of capital: the first applies economies of scale in reducing marginal cost of capital as the power plant's capacity increases.

Nonetheless, implementing a single power plant's production can be determined assuming a series of constrains that goes from the type of energy market to local demand. As already discussed, PPAs can be a constrain as well to strategic decision in production: in that case, the price negotiated depends on the LCOE which is influenced by the existence of penalties, especially minimum penalties, and the combination of maximum and minimum ones. It is not possible to consider economies of scale in energy production as they are considered in a common goods and services market: the existence of physical, technical, and legal constrains that characterize energy markets can make mostly inconvenient to rely on implementing the production strategies to make the marginal costs to drop.

[15] An entire LCA is based on a functional unit, which a quantified description of the performance of the production system, which means defining exactly which are the limits of the analyzed system. Even if we are considering the same quantity of CO2 produced per KWh/h, geographical, technical, financial factor can intervene in making the analysis not comparable.

Another possibility is focusing on learning rates: this is again related to the concept of *learning by doing* and not simply implementing the production of a single power plant but increasing the number of similar technological power plants producing in the market. The longer the technology is in the market, the more cost reduction is achieved thanks to technological improvements and acquired competences.

1.2.5 LCOE for grid parity determination

Commonly related to the LCOE formula is the concept of *grid parity*. The grid parity of an alternative source of electricity is reached when its LCOE is lower or equal to the price of electricity on the market. This is considered the rule of thumb of whether a technology is marketable or not, concerning its competitive capacity against fossil fueled power plants. Nissen and Harfst [6] demonstrated that using grid parity to guide investment choices in technology could be misleading, since a technology con be competitive even if its LCOE is above the electricity market price.

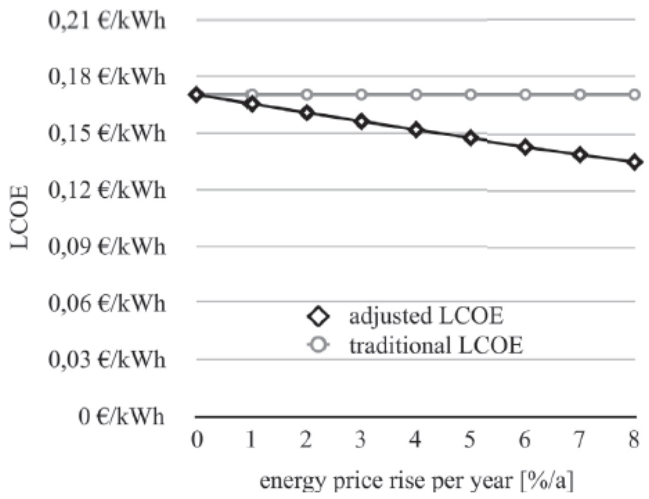
The authors proposed a reviewed model to better define market dynamics concerning technological competitivity. They start from redefining the LCOE in terms of *Net Present Value* (NPV): as already discussed, the LCOE is a cost-benefit measure, so that the idea of grid parity relies on the fact the benefits, in terms of electricity price, overcome the costs necessary to sustain the production. Reaching grid parity means determining the price p for energy, saved, or replaced, that equals the NPV. What it is problematic in the common assumption is that as every steady-state point, the equation is solved for a constant specific price. This means assuming that the electricity price is constant while comparing it to a measure, the LCOE, which relies on a time variable t, defining the time frame in which the power plant is operating.

In their work, they demonstrated a technology can generate a positive NPV and be considered profitable even if the LCOE is higher than the energy price on market. This was achieved assuming a 5% annual price rise for energy starting from a base price in $t=0$ of 0.15 €/kWh, which is still a simplification for reality, but that suits the experimental purposes for the study. A calculation spreadsheet was set for a hypothetic renewable technology, which a time horizon of 5 years, which is clearly unrealistic, but it suits the experimental purposes of

the exercise[16]. They obtained a NPV positive value of 2,266 €, while LCOE value of 0.17 €/kWh found is clearly higher that the starting price.

This contradiction led to the necessity to adapt the model to, at least partially, include the perspective of a non-stable price for energy. The *Energy price adjusted LCOE* includes a component to assess so, an *energy price rise rate* (epr): it means that in the LCOE formula are incorporated the expectations on an annual price variation of the energy price. On one hand, this is still a simplified solution to the complexity of energy market: it is well known that variations on the spot market for electricity happen daily, even hourly. The purpose of the authors, however, was trying to incorporate trends and expectation on the evolution of the market in a macroeconomic perspective.

Fig.3: LCOE depending on epr variations



*Source: Nissen and Harfst (2019), Shortcomings of the traditional "levelized cost of energy"[LCOE] for the determination of grid parity**

Including expectations means forecasting the possible scenarios for a more complete analysis: *Fig.3* shows the difference in LCOE values between assuming no variations in electricity annual prices versus varying epr from 0% to 8%. Since the component is added in the denominator of the equation, the more the epr increases, the more it is reflected on the electricity grid price and the more the LCOE drops. With this simple adjustment it was

[16] For any further explanation of the assumption made, see the original paper.

possible to adjust an unprecise measure for grid parity to a more complete index which allows to properly plan investments in renewable power plants according to how the market is expected to evolve in future years.

1.2.6 Final considerations about LCOE

So far, different strengths and weakness were discussed concerning on the LCOE measure and as many possible improvements were proposed. Summing up on what reported so far, it is possible to firstly say that the LCOE is the most complete measure built on a cost-benefit analysis for planning public national energy strategy. It is because it includes the wider set of costs accountable, it includes a discount factor which makes monetary values in different periods comparable, and it provides a unitary cost of energy which makes different technologies comparable.

The possible weaknesses are principally related to the high level of assumptions made for building the model. However, this is not uncommon in Economics: trying to describe reality through modelization necessary need assumptions that simplifies complex phenomena. Additionally, the strength of the LCOE can be considered an open model: further assumptions or implementations can be always done, as shown with the example of PPAs or a non-constant electricity price. Therefore, the LCOE was chosen in this analysis as a valid index to measure the energy cost even in future scenarios that will probably imply new variables and new hypothesis.

In this work, the original LCOE formula is taken as a starting point, but the newly added features are used for building model. It includes a WACC component, meaning that simulations can be done as well on discount rate, together with a Montecarlo simulation to simulate variations of the LCOE according to the experimental setup. Different estimation techniques will be compared to answer to new possible challenges and to answer to the evolution of the energy market.

2 Machine learning

Machine learning (ML) is a subfield of Artificial Intelligence (AI) that study the way in which the computer systems can perform a specific task without using explicit instructions, relying on patterns and inference instead. In practice, algorithms (the so-called *learner*) build a mathematical model based on sample data, known as "training data", to make predictions or decisions without being explicitly programmed to perform the task.
A complete overview of the ML areas is very hard to be made but some introductory works could be useful to approach this topic from the fundamentals [**12-17**].

2.1 Some historical notes

The term machine learning (ML) is nowadays part of the common talking so deeply to become a slogan. ML is often compared to inferential statistic or econometrics, but, since is inspired to replicate the human computational process is, indeed, radically different in concept, although like the previous ones especially in executing single tasks (like pattern recognition). ML is the heart of the AI, so few notes on history of the AI will be provided, to better understand what ML is really is.

Additionally, the techniques used in this research will be introduced: Multilayer Perceptron Neural Networks (MLP) [**18-23**], Support Vector Machines (SVM) [**24-29**] and Decision Trees [**30-36**].

Subsequently, a case study in energy cost calculation will be provided as an introduction to the experiment presented in the next section.

2.1.1 The Early Ages

The story of AI is normally indicated in the 50s, when the technological development, together with Sci-Fi novels, started to make the idea of an intelligent robot something familiar to the most. Alan Turing[17] was one of the most emblematic figures behind the theorization of

[17] Alan Turing was an English mathematician who dedicated his career to theorizing the mathematic framework that would have become the base of modern computer science. Turing is very popular for many good reasons,

AI: one of his most famous works [**37**] starts with the emblematic question "*Can machines think?*". Instead of directly answering the question and focusing on the difficult task of defining the concepts of machine and intelligence, he proposed a game, called "The imitation game". The Imitation game must be played by three people: a man (A), a woman (B) and third subject, of which the gender is irrelevant, that will play the role of the interrogator (C) and that is called to ask questions to the subjects. The interrogator cannot see the other players and classifies them as X and Y: the task is to declare at the end of the game which one of X and Y is either A or B. The perfect environment for playing the game is the one where the interrogator's questions are answered by typewritten answers, so that voices cannot influence the investigator's judgment. The question proposed by Turing to the original one is what happens if one of the players, for example A, is substituted by a machine which has the capabilities to answer? Will the investigator be able to define which one is the machine and which one is the human being? If the investigator is uncertain on distinguishing the two entities, it is possible to affirm that the machine is performing at the same level as a human being and that, to the extent of the task, is showing a form of intelligence that is like ours.

Nowadays, it is not uncommon to think about automatized answering systems: more and more companies provide an automatized costumer services systems, where the assistance to the public is provided by an AI. It is possible to affirm that the Turing test is verified at least for easier and more "mechanical" tasks, while for more complex topics it is not: AI are able to make a small talk with a human, understanding the context, but it could be harder when it comes in having a conversation with an expert on an extremely specific subject. At the time, when Turing was theorizing this innovative way to define intelligence related to machines, technology was not at the state for which something like that could be proved. The greatest limit to verify the hypothesis was due to technological state of computers: the machines that were used at the time were way far from any form of intelligence, since they can only execute commands, without being able to store them. It is possible to say that they were lacking on the key skill that will define intelligence, which is the capacity of "learning".

Even if there were at the time physical limitation to reach completely what Turing prospected, a first attempt of an "intelligent" machine was proposed by Allen Newell, Herbert

---

one of which is that in 1939, he gave a fundamental contribute to decoding Enigma, the machine used by Germans to crypt the communications.

Simon, and John Clifford Shaw[18], who was in charge in writing the codes and developing the program in 1955: the Logic Theorist was a program designed to reproduce a problem-solving scheme as similar as possible to the one applied by a person, which practically makes it to be considered by many the first artificial intelligence program ever developed. As Newell and Simon explains (1956), the application of the program was on proving theorems presented in Whitehead and Russell's Principia Mathematica (1910) in the form of propositional logic. Logic Theorist was able to prove 38 of the 52 theorems presented in Chapter 2, but the most surprising fact was that the proofs provided were even more detailed than the original ones proposed by the authors. In Simon's words, *"[We] invented a computer program capable of thinking non-numerically, and thereby solved the venerable mind-body problem, explaining how a system composed of matter can have the properties of mind.*": it was concretely possible to think about a machine that could be part of the Imitation game and concretely confusing the investigator.

In 1956, it was presented to the public in the Dartmouth Summer Research Project on Artificial Intelligence (DSRPAI), hosted by John McCarthy and Marvin Minsky [**38**][19]: it was the first time in which the phrase "artificial intelligence" was used for a scientific event. Even if historically the conference can be considered a milestone for the AI, in facts it was not possible to gain any relevant scientific result from the event since the participants were not able to agree on a shared framework of standardizing methods for the field. However, the codes wrote by Shaw to transfer the mathematic theorization of Logic Theorist on a computer were the bases for developing the IPL (Information Processing Language) and McCarthy adopted the same symbols to develop the LISP Programming language, which is still discussed by AI researchers.

2.1.2 Eliza, the first chatbot

Regardless the DSRPAI actual results, the event can be considered the starting point for the growing interest in AI that characterized the following years. The effort was focused on replicating forms of intelligence as close as possible to human's ones, through heuristic

[18] At the time of the invention, Herbert Simon was a consultant at Research and Development (RAND) Corporation while Allen Newell was a scientist studying logistics and organization theory in the same company: together, they theorized the mathematical configuration of the Logic theory, while John Clifford Shaw was a RAND computer programmer who oversaw the coding for Logic Theorist.

[19] John McCarthy and Marvin Lee Minsky were two American cognitive and computer scientists, considered among the founders of the AI as a discipline.

programming[20] for example. Following the idea of the Turing test, chatbots, small software's able to emulate a conversation with a real person on a predetermined topic, became more and more popular: in 1966, ELIZA, a chatterbot reproducing the behavior of a Rogerian psychotherapist, was invented by Joseph Weizenbaum[21]. In addition to ELIZA's ability to replicate human conversational skills, another interesting fact is that its script, keywords, and their associated transformation rules, was data [**39**] which means that it was not part of the program itself so that ELIZA is not restricted to any particular response or language, but it could recognize and answer a conversation in Welsh, German and English.

This step was fundamental: until ELIZA, the capacity of storing and sorting information were not a prerogative of machines, which had the primary use of executing a command on the base of the inputs provided. With the separation of the process from the data, ELIZA was able to show a primitive form of "intelligence" since it was part of the machine's duties to choose from the data which information was suitable for the ongoing conversation. The process is like what modern algorithms do, but it was extremely innovative at time, considering also computational capacities of computers in that age.

Technological limitations were of the greatest obstacles for further development of the AI in later years: ELIZA demonstrated that the key ability for a machine to resemble intelligence was relying on data, but computers could not store enough information as well as process it as fast as necessary to have a proper response to the task. In the '70s, most of the scientists working on AI were able to think and even mathematically theorize machines with the general intelligence of the human beings, but practical constrains do not allowed it. That decade, however, was crucial for the implementation of chess players programs. Claude Shannon[22] was the first who wrote a paper about developing a chess playing program in 1950 [**40**], describing two principal approaches: Type-A programs, working with a min-max search algorithm and computing thousands of moves, or Type-B, working with strategic and heuristic approach.

[20] The term "heuristics" indicates the shortcuts and mental strategies that human beings put into action whether they are involved in pursuing a goal. Heuristic programming tries to infuse the same logics and behavior into a machine in problem-solving activities.

[21] Joseph Weizenbaum was a German American computer scientist and professor at Massachusetts Institute of Technology (MIT).

[22] Claude Shannon was an American mathematician, electrical engineer, and cryptographer known as "the father of information theory", which is the scientific study of the quantification, storage, and communication of digital information.

During the ‘50s and the ‘60s, Type-B programs were mostly developed, but in 1973 the developers of the “Chess series program” switched to Type-A programs: Type-Bs were preferred because of the simpler conformation and the need of less input data to work, but they could easily behave wrongly following an inappropriate rule of thumb during a match. The technological capabilities available at the time, even if limited, made possible to focus on Type-A programs, the ones using “brute force”.

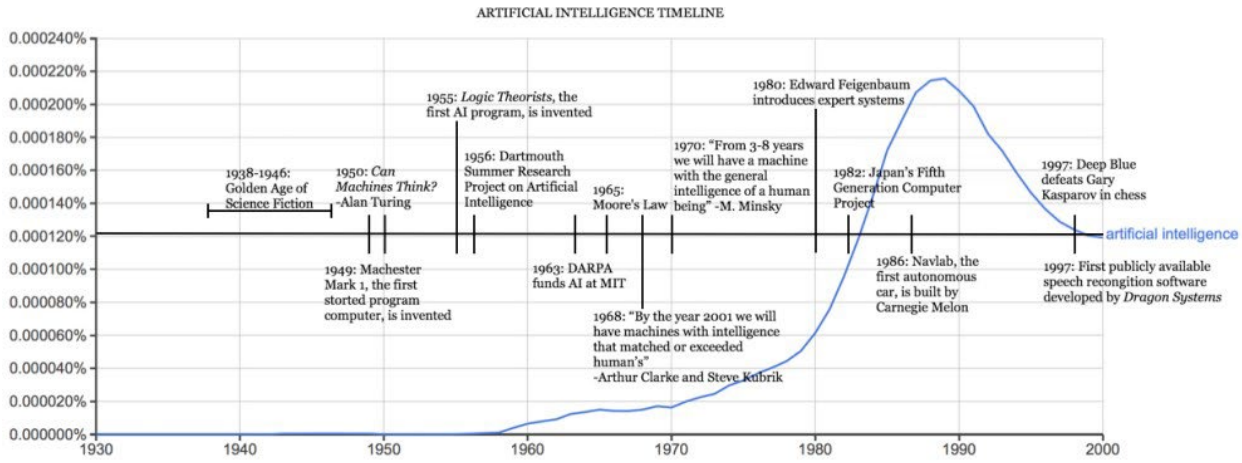
2.1.3 From the ‘80s to the “big data” age

The 1980s were a fruitful time for AI: John Hopfield and David Rumelhart[23] [**41-43**] made the deep learning methods, which will be presented more in details later, popular and Edward Feigenbaum presented expert systems, intelligent computers that use knowledge and procedures to solve problems that are difficult enough to require a certain expertise in the solving process (Feigenbaum and Klahr, 2003) [**44**]. It was during the 1990s that AI innovation made great strides: in 1997, a Type-A chess player program, Deep Blue, developed by the corporation IBM, was able to defeat the world chess champion Gary Kasparov.

Deep Blue was able to compute around 200 million positions a second and averaged 8-12 play search depth, while humans can only consider near 50 moves to various depths: the victory was possible thanks to this extraordinary computational capacity, if Deep Blue were a Type-B program then perhaps the win would have been more interesting for the considerations on machine intelligence (Smith, 2006) [**45**]. 20 years later, in 2017, Google’s Alpha Go defeated the Chinese Go champion, Ke Jie: the greatest capability of Alpha Go was studying older matches and playing thousands of games against itself, thanks to improved storing capacity, a greater number of data accessible and faster computational capability.

Fig.4: Timeline of the AI development

[23] John Hopfield is an American scientist most widely known for his invention of an associative neural network in 1982, David Rumelhart was an American psychologist who made many contributions to the formal analysis of human cognition, through mathematical psychology and logical reasoning.



Source: Anyoha (2017), The History of Artificial Intelligence [**46**]

The invention of Google in 1998 can be considered the starting point of the era of the "big data": constant flow of data continuously processed that provides algorithms with all the necessary information to train themselves. The 21st century is characterized by important changes in information technologies, like the Internet of Things (IoT), cloud computing and social networks: these three key components increased drastically the speed at which data are generated and accumulated [**47**]. Thanks to the IoT and social networks, the type of data that are processed nowadays are profoundly different from the script used by ELIZA: data production is constant, interconnected and deeply detailed. The social network era started with the foundation of Facebook in 2004: the original idea was merely related to digitally archiving basic information and pictures of Harvard students, while todays Facebook accounts for more than 2.2 billion monthly users.

Nowadays, Facebook is just one of the social networks available, but it was a pioneer in using big data to provide a customized experience to the users. The data produced by social networks users are so highly detailed that from a commercial point of view a mass personalization can be provided (Peters, 2012) [**48**]. The greater share of Facebook's revenues comes from advertising revenues thanks to the possibility of marketers to propose targeted advertisements based on the data produced by subjects through their Facebook profiles [**49**].

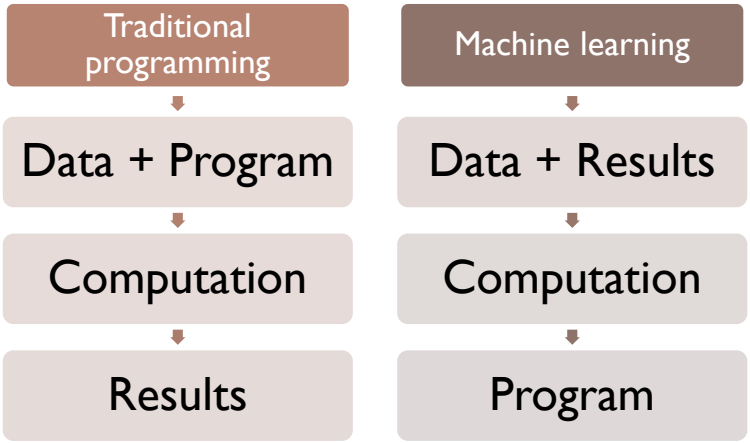
The idea of interconnection is also relevant in IoT: the term was coined in 1999 and it refers to idea of transforming everyday objects into smart devices being connected to the network and able to gather and share data (Trend Micro). In IoT, data production is possible without human intervention thanks to the object's sensors from the environment: a smart air

conditioner system can automatically blow cooling air when the room's temperature reaches a certain value. Social networks and IoT, in addition to the data production and usage, have in common the presence of an algorithm, executing a predefined task thanks to the data provided.

2.2 What is "machine learning"

Most of the automatized processes replicate the exact same pattern: gathering data, processing the data acquired and outlining the information necessary and finally using that information for executing the assigned task, improving the performance through experience. The process described can be achieved through a machine learning process. Before introducing ML techniques, it is necessary to provide a proper definition of ML, as the one form IBM: "Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy." ML is a subset of AI that relies on the utilization of data for making decisions from what observed from the data itself: it involves the same components of traditional programming, but what differs is the way in which those components are combined. In traditional programming, the machine is directly programmed and from the computation on data is asked to provide results; ML starts from data and results and uses computation to come up with rules [**50**].

Fig.5: Traditional programming vs Machine learning



ML is often used in forecasting as an alternative to classic statistical tools, because of the possibility to model more complex phenomena, difficult to be described a priori. The process is like the switch from traditional programming to ML: in econometrics, for example, a certain statical model, a linear regression for example, is applied to data and results are commented. This implies that the researcher has a prior idea of the relationship within the input variables, which usually comes from scientific literature. However, inputs interactions can be hidden or unclear: ML algorithms can discover these unknown patterns and using them for improving the computation. Domingos [**51**] defines ML as "doping applied to scientific method": the hypothesis testing process is the same one always applied in the scientific method, but thanks to ML it is possible to test simultaneously multiple hypothesis in a second, while a scientist would have spent their entire career working on them.

The role that the scientist has in the process depends on the type of machine learning classifiers used and what is the purpose of the analysis. ML techniques refers to two main categories with two primary scopes: supervised learning for building predictive models, and unsupervised learning for constructing descriptive models. In supervised learning, labeled data is used to train the algorithm: the algorithm has the aim to adjust the weights attributed to each information gained to avoid overfitting or underfitting of the model. Predictive models are obtained by supervised learning: a target feature, the one to be predicted, is specified to the learner and its task will be to discover the relationship among the target feature and the other features from the dataset. The term "supervised" does not imply an active role of the scientist in the process: the only human intervention in the learning process relates to correctly specify the target feature, providing the algorithm with the instructions for the task it must learn.

Predictive models can be split into two categories: classification models and numeric predictions. When the target feature can be expressed as a categorical feature, a class, divided into levels, classification models are used to predict whether a certain state of the world will happen. An easy example is antispam filter: the algorithm is asked to classify an email as spam or not spam, according to the given criteria provided by human supervision. If the target feature can be defined as a number, numeric prediction is suitable: economic variables, as the average income of a certain area can be predicted from significant inputs provided. This is the specific case for Economics in which ML techniques can be a valid alternative to Econometrics.

The other type of learners is unsupervised learning: in this case, there is no necessity to establish the target of interested, since every feature is as important as the others. In unsupervised learning, the algorithm is asked to search for patterns or data groupings exploring the given data: the greater the number of inputs provided, the greater the possibility for the algorithm to discover hidden relationships among variables. Unsupervised learning is often used in data mining[24], which aims to extract useful information from big data available concerning a certain topic: pattern discovery, for example, is often used for market analysis and costumer segmentation, so that costumer preferences are directly revealed from the purchasing data that they directly produce. Another possible use is clustering, which final aim is dividing the data in homogeneous groups: the ML algorithm can divide into groups, but it a human's prerogative to find the common characteristic for which each group was created by the learner.

It is also possible to classify existing ML algorithms according to the idea of intelligence that was infused in the machines while developing them: even if it seems more a philosophical than a technical concern, it influenced the way in which algorithms perform in solving the assigned task. In Pedro Domingos' book, [**51**], the author underline five "tribes" of machine learning, according to the principles supported by each group. The main idea supported into the book is if and how will be possible to find "the ultimate learning machine", an algorithm concretely able to process any sort of information, learn from it and using it to take decisions or complete a task without any necessary support from the human beings: this idea of Master Algorithm corresponds to the artificial general intelligence (AGI)[25].

Going deep in Domingo's argument, the first tribe is constituted by the Symbolists, which firmly believe in a type of logic intelligence, as the one used in mathematical logic: the set of rules used in algorithms is the "if-then" type of logic, which effective, but that can result as rigid when decisions are taken in uncertain conditions. The second is the Connectionists, who tried to model algorithms on the same patterns followed by the human brain: the idea of connections relates to the connections that happens between neurons when signals pass from one to another. The Bayesians focuses on uncertainty and probabilistic values: ML is seen as a form of probabilistic inference where every possible outcome happens

[24] As we found in Lantz [**52**], the term data mining is erroneously used as a synonym of machine learning. Machine learning can be applied to tasks different from data mining, like the ones concerning predictive models, while if data mining methods are used, it is highly likable that ML is used.

[25] AGI will correspond in the final step of ML, where a machine will be able to understand, learn and replicate any cognitive action that a human being can perform. Another step further could be the surpass of the human intellectual capacity from an AI, ending in the Artificial Super Intelligence (ASI).

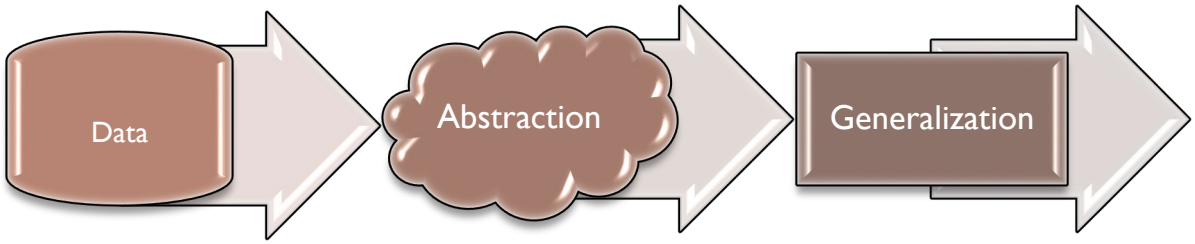
with a certain chance. Evolutionaries refer to natural selection as the key for learning: the focus is on the process, focusing on steps and iterations, miming what the natural evolutions of species have done with their cognitive abilities. Lastly, Analogizers base learning activity on the capacity of finding similarities, evaluating the level of similarity between two different units. This classification could appear purely theoretical, but it provides a clear explanation of the logic behind different type of algorithms: decision trees are a typical decision-making tool for Symbolists, while support vector machines (SVM) can be attributed to the Analogizers approach.

2.2.1 How machine learning works

Regardless differences in the logic applied, the machine learning workflow is a common process for every algorithm. Lantz (2019) reduces the learning process into three essential components: data input, abstraction, and generalization. Data input is the shared starting point: as already explained, data can be labeled or not depending on which type of learning process is in act (supervised or unsupervised). In the abstraction phase, data is organized according to the meaning that can be attributed to it. Data itself is composed by numeric observations with no specific meaning, stored in the computer's memory as a sequence of zeros and one: abstraction is necessary for specifying the meaning of the data, connecting its numeric representation with the reality which generated it. The knowledge representation, the formation of logical structures that transforms raw sensory information into meaningful insights, is based on these abstracted connections: practically, the algorithm organizes input data into a model for extracting the descriptive pattern of it. Modelization can be practiced through equations, diagrams, logical if-else rules, and clustering: the model choice is subject to the purpose of the analysis[26]. The model optimization process applied to data is called training: training the machine to describe the data is the first step of the learning process.

Fig.6: From data, through Abstraction, to Generalization

[26] The model choice concerns to the difference between using supervised learning for predictive scopes in contrast to using unsupervised learning for descriptive purposes.



Source: Lantz (2019), Machine Learning with R

A step further is generalization: the learner must be able to apply the abstracted knowledge gained in training on a similar problem. Generalization implies that the machine can understand the key features of the problem, instead of examining each single information provided: this is related to the concept of heuristics, which was already introduced, mental shortcuts that human beings enact while completing a task. To verify if the abstraction and generalization process was successful, testing is necessary: the algorithm is asked to apply what it was learnt in the previous steps to a new dataset, performing the same task as the one it was asked to conduct on the training one. Referring again to the antispam filter example, in the training process, the learner was taught to separate a spam email from a non-spam one following certain criteria, for example identifying key words. In the testing phase, the anti-spam filter is provided with a new set of emails: the more emails are correctly identified, the better the learner is performing.

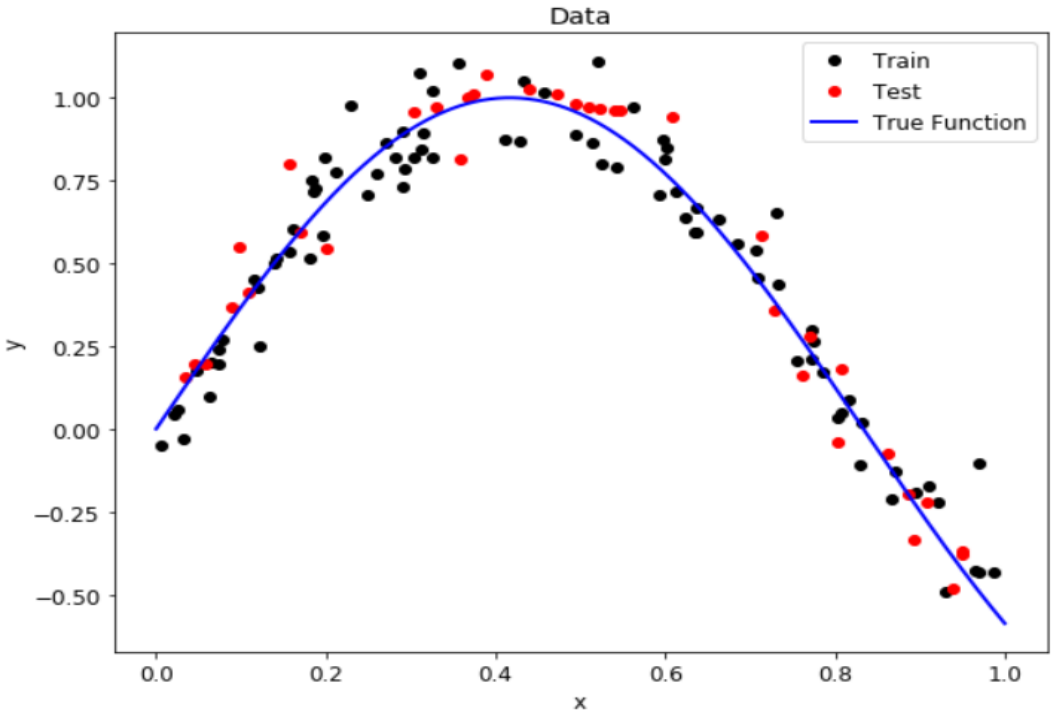
The advantage of machine-learning models over traditional statistical models is their ability to quickly consume enormous numbers of records and thereby more accurately make predictions. However, this does not mean that ML models are infallible: leaners' capability of prediction is based on the quality of data they have been trained on. If the data is somehow biased, so it will be the decision making of the learner. If data incorporates a certain bias, in the training process, the algorithm will incorporate di bias in the learning process, and it will apply a biased rule in the testing phase when it is asked to perform its task. Biases are particularly relevant in data which was generated by human beings: an example is automatized credit scoring processes where the algorithm was trained on time series data of the previous credit decision performed by bank clerks[27]. A more detailed analysis about biases implication in decision making will be provided in the next chapter.

[27] Every cognitive bias applied in granting or not loans to certain subjects by the bank clerks is incorporated in the data and the learner will be trained on associating certain characteristics to liability. Clarifying, if the bank clerk was racist so will be the algorithm.

Another implication of the close relationship between learning and data quality is the underfitting vs overfitting problem. In the training phase, the algorithm is asked to model the relationship between the inputs, the features, and outputs, the labels; in the testing phase the learner is provided only with the features and asked to make the predictions about labels. Data is not completely able to fit a real-world trend: there will always be unmeasured relationships or noise in the variables considered in the model. There are several reasons for noise in data: it goes from simple measurement errors, concerning the classic wrongly reported data, or even errors in how data are stored, like missing values because of incorrectly coded or corrupted values. The data itself has its own trend: the "true function" represents the statistical distribution of it. The first problem concerns the fact that the model generated by the algorithm is not flexible enough to represent the data provided: in the case of underfitting, the variance is low, and the bias is high.

What happens is that the learner, in its process of modulization, oversimplifies the relationship between the inputs, describing it as less articulate than how concretely is. When it comes to the testing phase, the error rate is high because of the model incapacity of shape the complexity of data. It is possible to increase the capability of the model through an additional step, which concerns improving the model performance: what usually happens is that cost function is developed to make some errors more relevant than others. If we are asking a classifier model to determine whether a mushroom is poisoning or not, the situation in which the model classifies as poisoning a non-poisoning mushroom should be weighted more than the one in which a non-poisoning one is classified as poisoning. Eating a poisoning mushroom will make someone die while not eating an edible mushroom will not affect that much anyone.

Fig.7: The overfitting problem



Source: Koehrsen (2018), Overfitting vs. Underfitting: A Complete Example [**53**]

This example can seem out of context, but it can be easily applied in more complex situation, like choosing to finance or not a certain project because of the probability of default. The risk that comes from the process of improving the model performance, and therefore trying to incorporate the most possible data noise in the model, ends in overfitting. Overfitting makes the model to goodly perform in training but poorly in testing phase because the learner is not able to generalize. The algorithm perfectly learned to shape the training data but did not extrapolate the existing dynamics that connects the variables: it ends with a model not able to be use for predictive purposes. It is important that in the improving phase to balance between accuracy and flexibility.

The process described, together with the possible implications, it is commonly shared with all the learners, regardless the purpose, the type of learning and the logical structure. A possible difference can occur in the level of intelligibility of the various steps. The learners used in this chapter are commonly referred as black box methods: in black box methods the mechanism for which the input is transformed into the output cannot be easily descried or it cannot be described at all.

3 Data and Methods

3.1 Data

The used data come from IEA handbook about solar photovoltaic technology for Italy, in particular data are referred to a plant of 0.83MW, as showed in table 1 [**4**][28].
Instead to use a fixed value for discount rate, like IEA, we've used the WACC: the used value, reported in table 1, reflect the calibration operates respect the LCOE value calculated by IEA.

Variable	**units of measure**	**Value**
Cost of capital	U.S. Dollar/MWh	21,77
Power	MW	0,83
Average Load Factor	%	27%
Average Auxiliary Factor	%	100%
Auxiliary Power	%	0%
Lifetime	Years	25,00
Weight of Debt	%	50%
Interest rate of Debt	%	5%
Taxation Rate	%	30%
Risk Free Rate	%	1%
Expected Market Risk Premium	%	2%
Beta	%	100%
Construction Time	Years	1
Operation and Maintenance Cost	U.S. Dollar/MWh	27,85
Cost of Fuel	U.S. Dollar/MWh	0,00
CO2 Price	U.S. Dollar/Ton of CO2	30,00
Decommissioning Cost	U.S. Dollar/MWh	2,51

Since the data was not sufficient for to test the predictive power of the different models used in the work, a numerical simulation based on such data was performed. The first kind of simulation involved just a predefined variance (± 10%) arbitrarily imposed to the IEA calibration values.

[28] See at page 60-61.

The simulation was performed using 10.000 replies of each parameter of LCOE model, then using for training phase of ML models the first 1.000 replies, testing results on the subsequently 9.000 observations.

3.2 Methodology

Linear Model

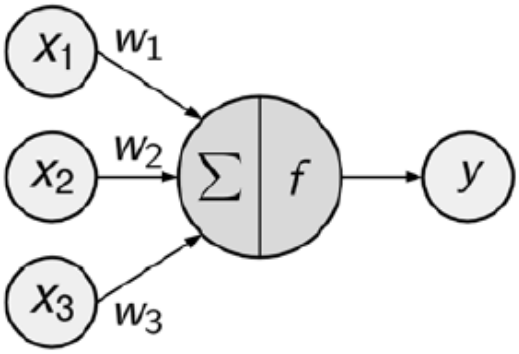
Due to the massive abundance of excellent free resources related to this topic, is not intended by the authors to bother the readers proposing for the umpteenth time a well-known topic so we just suggest a looking to some classic to refresh the basis [**54**], if this is the case.
Furthermore, the work used as a baseline for the performed test, is also a practical and useful resource to quickly remind the essential to understand the simulation [**52**][29].

Neural networks

Starting with *Artificial Neural Networks* (ANNs), they try to shape the relationship between the input and the output with the same processes that a human brain would have used. Neural networks were particularly popular both in neuroscience and computer science during the '60s, they had an eclipse during the '70s and had a new era in '80s: nowadays, they are widely implemented in different tasks because of their vast potential. The first learning processes of ANNs involved simple logical constructs like the logical *AND* function or the logical *OR*, which was necessary to try to replicate as realistically as possible the brain's functionality. In ANNs, the brain's network made by the interconnection of neurons is replicate thanks to *artificial neurons* or nodes to achieve the learning tasks. A neural network consists in sets of consecutive layers made of nodes sets, transmitting the signal for each layer, ending into the output signal, the result. Each node signal is weighted according to the importance that it has in determining the output. The signal is passed on different layers according to an activation function *f*, so that the output is determined by the activation function value of the weighted sum of the inputs.

Fig.8: The activation function

[29] See chapter 6.



Source: Lantz (2013), Machine Learning with R

There are different existing types of neural networks that can be used according to the complexity of the problem to solve, and they can be distinguished according to three factors: the type of activation function chosen, the architecture of the network and the training algorithm. Through the activation function, the mechanism for which information is transmitted from one neuron to another is established. It involves setting a threshold that establish whether the information should be transmitted to another neuron or not: passing or not the threshold depends on the value that the activation function attributes to the total input signal.

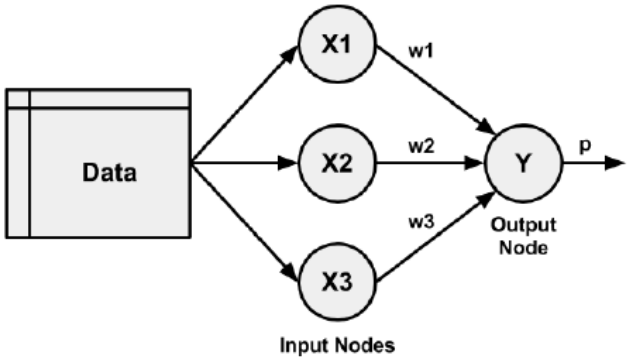
Thinking more concretely, what happens is that a set of information is available to determine a certain output: not all the information is necessarily needed to determine the output or either the level of information for a certain input is not enough to concretely contribute to the computation of the output. In those cases, the threshold is not reached, and the neuron is not activated. One of the most used activation function forms is the *sigmoid activation function*[30], because of property of being differentiable: since it is possible to calculate the derivate across the entire range of inputs, it smooths the threshold calculation for every neuron.

The *architecture* of the network concerns the number of neurons for each layer, how many layers are included, the way in which the information travels between the neurons[31]. The number of neurons in each layer is closely connected to the type of data: more neurons for each layer means a greater descriptive capacity of the data, together with an increased risk of overfitting. Additionally, an overstructured ANN is slower both in training and in

[30] Different functional forms can be used to define the activation function: the simplest one is the unit step activation function, but it come also in linear, satured linear, hyperbolic tangent and gaussian form. The sigmoid will be taken as reference in this work.
[31] Concerning if it is possible for information to travel back and forward in the network.

computing: a balance between computational power and efficiency is fundamental. The number of layers relates to how many numbers of input nodes are present in the network: again, for each neuron, a connection weight is associated. The elementary architecture of an ANN is a *single-layer network*: one set of input neurons, associated to one set of weights, obtaining an output.

Fig.9: ANN with a single layer structure



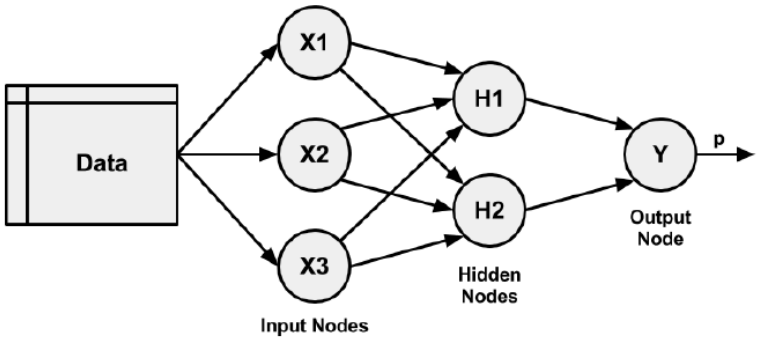
Source: Lantz (2013), Machine Learning with R

This type of network is well performing for elementary pattern classification tasks, but not for more articulated purposes. *Multilayer networks* are made by adding to the input nodes' set one or more hidden layers, rising the computational capacity of the algorithm. The complexity of the network is directly proportional to the complexity of the task itself. The relationship between input nodes and output nodes passing through hidden layers cannot be explained: the method is a black box one because it is not possible to describe the connection the network builds neither to count how many neurons are displayed in the hidden layers. The number of layers displayed relates to the "*depth*" of the model: the more layers contribute to computation, the deeper the network is. The term *deep learning* is often associated directly with neural networks[32], however not all the ANNs are deep learners: a neural network with more than three layers can be considered a deep learning algorithm, while an ANN with only three or two is a basic neural network. This does not imply that the model is not efficient for

[32] As happens with data mining, deep learning is also used as a synonym for ML, while the terms are not interchangeable. As described by IBM (2020), ML, deep learning and neural networks are all subfield of AI, deep learning is a subfield of ML, and ANNs are a subfield of deep learning, considering again that not all ANNs models are deep learners.

the prospected purpose, but that the problem's structure does not require additional complexity.

Fig.10: ANN with multiple layers structure



Source: Lantz (2013), Machine Learning with R

Lastly, ANN models vary according to the directions in which information travels within the networks' layers. In *feedforward networks*, input signal moves through the network in one direction, from the first input layer, through the hidden layers, if present, ending in the output. In *recurrent networks*, loops are allowed to let the signal to travel in both directions. The fact that information travels in a straight direction does not imply that in feedforward networks information cannot travel back. This relates to the type of logical structure used to train and adjust ANN results, called *backpropagation*.

Backpropagation stands for "*backward propagation of errors*": it is an algorithm for supervised learning of artificial neural networks based on *gradient descent*. The process involves the error function of the ANN and calculates the gradient of the error function according to the neural network's weights. Practically, what happen is that the weights in the network are given randomly in the first step, since there is no prior existing knowledge about the relationship between input and output. In the *forward phase*, the neurons are activated in sequence according to their weights, layer by layer, ending in the output. In the *backward phase*, the results obtained from the forward phase are compared with the task expectations, so that the difference between the network's output and the true value results provides with an error value that can be back propagated to balance weights and connection between neurons reducing future errors. The way in which the algorithm attempts to minimize the error through the entire network is considering the derivative of each neuron's activation function to

understand the direction in which the weights should be balanced according to the error: this technique is the gradient descent.

The type of ANN that will be used in this work is a *Multilayer Perceptron* (MLP): it is a type of network that uses a sigmoid activation function, it is part of the multilayer feedforward networks family, its performance is improved thanks to backpropagation, and it can be easily adapted to numeric prediction. In this specific case, it will be applied to a regression for forecasting purposes.

Support Vector Machine

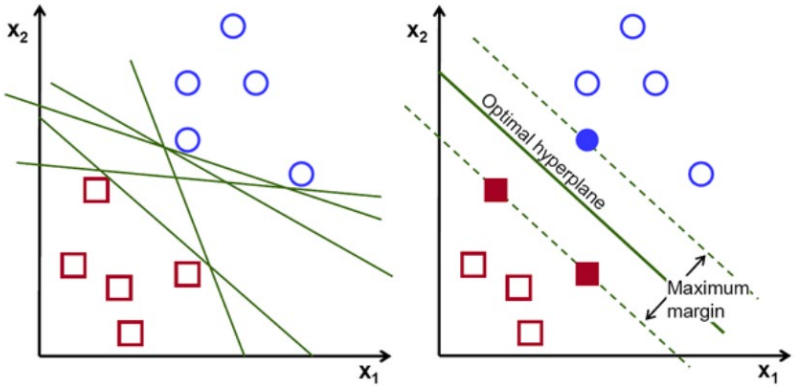
The second technique involved is *Support Vector Machine* (SVM). SVMs are powerful tools for classification which well-perform in numeric prediction: the main task for the algorithm is to find a *hyperplane* in a N-dimensional space. N stands for the number of features included, which allows to separate the data point according to a similarity principle applied to the observations. The hyperplane can be considered as a decision confine which separates the data in as homogeneous as possible groups: points that falls on either side of the hyperplane are considered belonging to the same class. The easiest case for the SVM to be operating whit is the case where the data is directly *linearly separable*, which is the case in which, regardless the disposition of data in the considered space, the objects can be separated through a simple straight line[33].

The algorithm solves an optimization problem: there is more than one possible separating line applicable to the data, but the SVM is involved in defining the *Maximum Margin Hyperplane* (MMH), which is the line that generates the maximum separation space between data. The result of this optimization problem is the maximum distance between the objects of different categories, so that the optimal hyperplane is defined. The MMH is defined thanks to the *support vectors*, which are for any category involved in the classification, the closest points to the MMH. The support vectors define the orientation and the position of the MMH in the space and helps to maximize the margin of the classifier, ending with a maximum margin. Each class of data must have at least one observation that works as a

[33] Hyperplanes are commonly defined in two dimensions; therefore, a straight line is the objected used to separate data, however this relates to the fact that it is harder for the human abstractive capacity to imagine data distributed in a 3D space. The hyperplane can be easily adapted to a multidimensional space imagining considering a flat surface instead of a line as data separator.

support vector: the algorithm applies vector geometry to identify the support vectors necessary to establish which of the possible hyperplanes among the set is the MMH. The method is a black box method because the optimization process is performed by the algorithm without any human intervention, so that what happens is specular to ANNs: the results come from a process that is not possible to follow or explain step by step.

Fig.10: The optimal hyperplane in an SVM



Source: Gandhi (2018), Support Vector Machine — Introduction to Machine Learning Algorithms

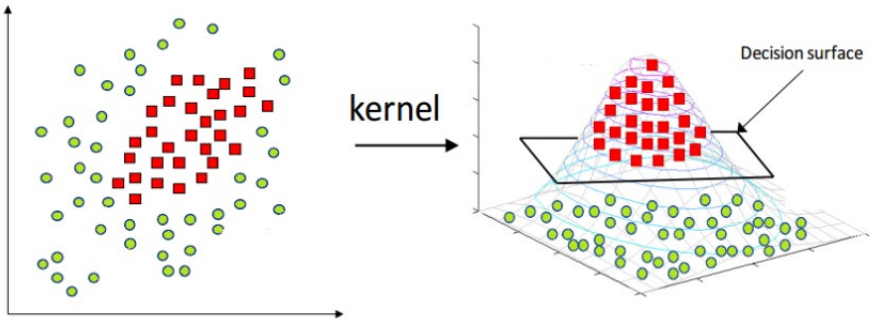
As already introduced, there are two main cases in which the data is disposed: linearly separable or non-linearly separable. The advantage of SVM is that even if it is a linear model for classification and regression, it can solve both linear and non-linear problems. For a simpler comprehension, binary classification will be assumed, so that the class considered are only two. In the case of linear separability, the MMH is simply the straight line as far away as possible from the outer boundaries of the data points' groups, which are called *convex hulls*. The maximum margin between the convex hulls is defined mathematically through quadratic optimization or involving Euclidean norm[34], which are anyways tasks performed by the algorithm. In the case of non-linear spaces, this methodology cannot be applied.

Moreover, the strength of the SVM is the visualization capacity of the data, which could diverge from the human capacity. The algorithm can apply the *Kernel trick*: the SVM can project the non-linear relationship of data in a different dimensional space. In this way, a

[34] The main purpose of this work is just provide a logical comprehension of what ML is and how to use it applying its techniques to a problem, in this case cost of energy computational analysis. For further explanation, the main reference book for this section is *Machine learning with R* (2016) by Brett Lantz.

linear relationship can be found and the MMH more easily defined, applying the already explained methodology. What the kernel trick does is adding new mathematical relationship to the data so that it can be linearly defined: the kernel trick allows to extend the computational analysis to a multidimensional level, discovering possible connections in data that are not clearly understandable in two dimensions.

Fig.11: The Kernel's trick



Source: Zhang (2018), What is the kernel trick? Why is it important?

A practical example of using SVM for performing regression, applied to financial forecasting is daily stock prices, where the price at time *t* is defined as a function of previous stock prices at time *t-1, t-2,…, t-k* [**55**]. Different methodologies are applied by technology corporations like IBM and Yahoo to perform the task and SVM is a robust technique to implement this type of research. The idea of applying SVM is ideal for a stock price forecasting for two reasons: the relationship within the price at time *t* and prices in previous year is of course not linear because of the great number of factors intervening and because of this computational complexity. Applying the kernel trick in computational analysis provides a wider look to relationship that it is not possible to model for human capacity itself because of computational constrains.

In other words, the change of perspective provided by the learner can outline connections that will be kept hidden otherwise. The strengths of a SVM are that they can perform a high accurate analysis without incurring in the risk of overfitting since it is a type of learner not excessively influenced by noise in data. Additionally, the complexity level of the black box method is even more high when kernel trick is applied, so that the results obtained are even more cryptic to interpret.

About Trees for numeric prediction we consider two categories:

Regression trees, introduced in the 1980s as part of the Classification and Regression Tree (CART)[35] algorithm: despite the name, regression trees do not use linear regression methods since they make predictions based on the average value of examples that reach a leaf;

The second type of trees is the model trees, born several years later than regression trees. About Model trees at each leaf, a multiple linear regression model is built from the examples reaching that node (model tree may build tens or even hundreds of such models). This may make model trees more difficult to understand than the equivalent regression tree, with the benefit that they may result in a more accurate model.

Trees that can perform numeric prediction offer a compelling yet often overlooked alternative to regression modeling.

Some pros and cons:

Pros

• the strengths of decision trees is combined with the ability to model numeric data

• Feature selection is automatic, very useful with a very large number of features

• To specify the model in advance is not necessary

• May fit some types of data much better than linear regression

• Does not require knowledge of statistics to interpret the model

Cons

• Not well-known and consolidated as linear regression

• A large amount of training data is required

• The overall net effect of individual features on the outcome is difficult to determine

• Some difficult of interpretation respect to a regression model can be arise.

[35] The CART algorithm is described in detail in [**56**].

In some cases, numeric decision trees offer distinct advantages:

decision trees may be better suited for tasks with many features or many complex, non-linear relationships among features and the outcome, these situations present challenges for regression;

regression modeling also makes assumptions about how numeric data are distributed that are often violated in real-world data (not true for trees).

Trees for numeric prediction are built in much the same way as they are for classification. Beginning at the root node, the data are partitioned using a divide and-conquer strategy according to the feature that will result in the greatest increase in homogeneity in the outcome after a split is performed.

In classification trees, you will recall that homogeneity is measured by entropy, which is undefined for numeric data.

For numeric decision trees, homogeneity can be measured by statistics such as variance, standard deviation, or absolute deviation from the mean. Depending on the tree growing algorithm used, the homogeneity measure may vary, but the principles are basically the same. A common splitting criterion is called the standard deviation reduction (SDR).

Results

This section proposes the results obtained running the four-model described in the methodology section. A dedicated R script was used to produces 10.000 replies of a Monte Carlo simulation, performed assuming that all the variables was described by a Uniform Continuous distribution. We've chosen to simulate the situation in which only significant parameter could be used to forecast LCOE, focusing on the Operation and Maintenance Cost, for two reasons:

1. the significance of the relationship between OM and LCOE for the considered technology
2. the relative abundance of information about status and perspective of such type of costs.

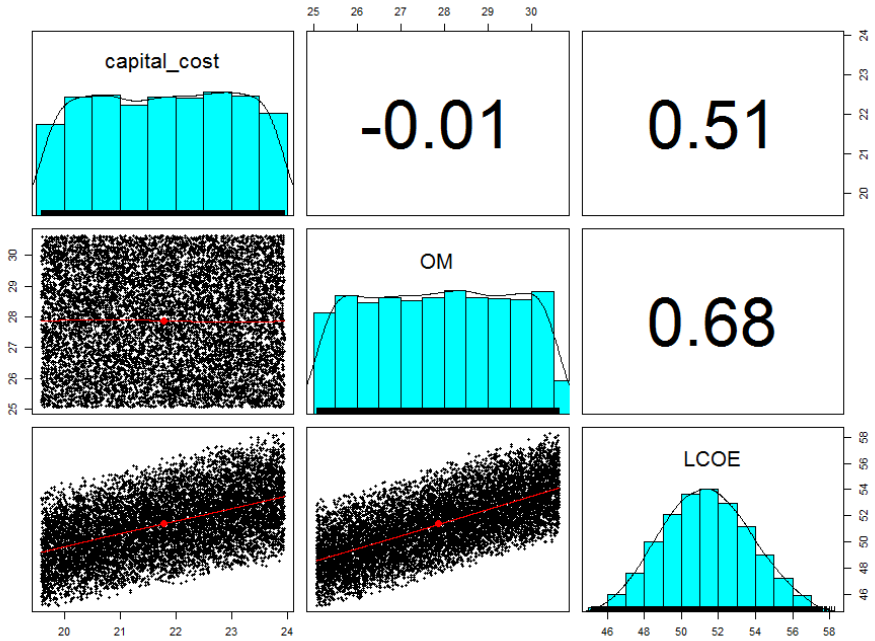
No improvements technique is proposed in this section: tons of details about them are literally Let's start for the first type of simulation.

Measuring forecasting accuracy - part I

Linear Model

An enhanced scatter plot can be created by using psych package, as shown in figure 12:

Fig.12: Results from estimation of LCOE by a linear model



Above the diagonal, the scatter plots have been replaced with a correlation matrix. On the diagonal, a histogram depicting the distribution of values for each feature is shown. Finally, the scatter plots below the diagonal now are presented with additional visual information.
The oval-shaped object on each scatter plot is a correlation ellipse. It provides a visualization of how strongly correlated the variables are. The dot at the center of the ellipse indicates the point of the mean value for the x axis variable and y axis variable. The correlation between the two variables is indicated by the shape of the ellipse; the more it is stretched, the stronger the correlation. An almost perfectly round oval, as with capital cost and OM, indicates a very weak correlation (in this case -0.01).

The output provides three keyways to evaluate the performance (fitness) of our model:

Residuals:

Min	1Q	Median	3Q	Max
-3.9868	-1.2721	-0.0686	1.2128	4.6451

The maximum error of 4.65 suggests that the model under-predicted expenses by nearly $5 for at least one observation. On the other hand, 50 percent of errors fall within the 1Q and 3Q values (the first and third quartile), so most predictions were between $1,21 over the true value and $1,21under the true value.

Coefficients:

	Estimate Std.	Error t	value Pr(>|t|)
(Intercept)	23.50593	0.30191	77.86 <2e-16 ***
OM	0.99994	0.01082	92.45 <2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

The stars indicate the predictive power of each feature in the model. The significance level (as listed by the Signif. codes in the footer) provides a measure of how likely the true coefficient is zero given the value of the estimate. The presence of three stars indicates a significance level of 0, which means that the feature is extremely unlikely to be unrelated to the dependent variable. A common practice is to use a significance level of 0.05 to denote a statistically significant variable. If the model had few features that were statistically significant, it may be

cause for concern, since it would indicate that our features are not very predictive of the outcome. Here, our model has OM as significant variables since it clearly related to the outcome in logical way.

Residual standard error:	1.74 on 9998 degrees of freedom
Multiple R-squared:	0.4609, Adjusted R-squared: 0.4608
F-statistic:	8547 on 1 and 9998 DF, p-value: < 2.2e-16

About the R-squared, a value of 0.46 is relatively good. Remember that also capital cost is a fundamental explanatory variable, and it wasn't included in the model to isolate the pure effect of OM in LCOE determination (since we are interested to forecast the LCOE with a single significant variable in our test).

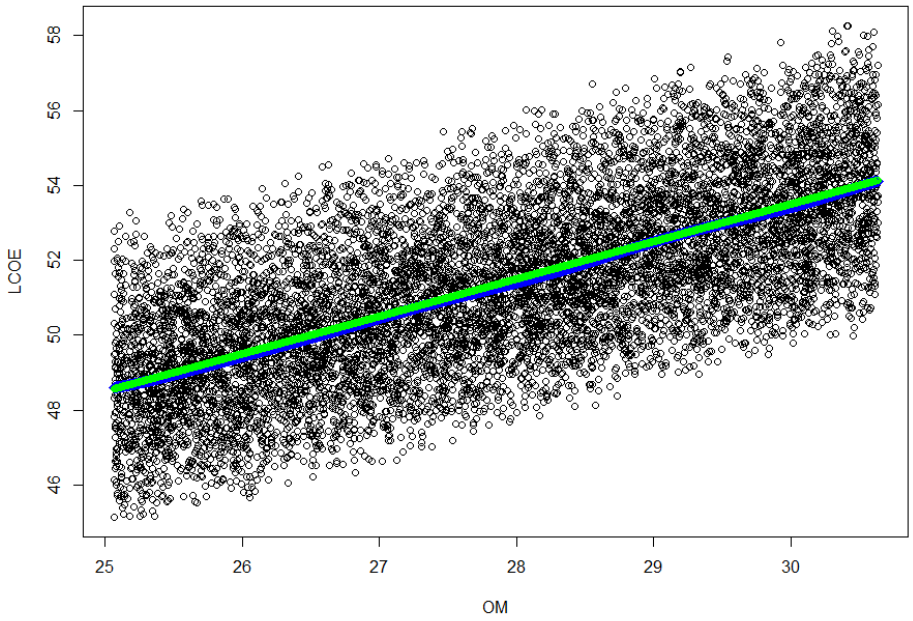
Artificial Neural Network - The Multi Layer Perceptron

Since we deal with a numeric prediction problem we can measure the correlation between our predicted LCOE and the true value with a result of 0.68.
We can also calculate the R-squared, obtaining a value of 0.46, namely, we get same results that the ones from the linear model.
This is not surprising since we have use neural network with a single hidden node that can be thought of as a distant cousin of the linear regression models : the weight between each input node and the hidden node is like the regression coefficients, and the weight for the bias term is like the intercept. In fact, the linear model previously used presents a correlation is 0.46.

Support Vector Machine

We've used the e1071 package from the Department of Statistics at the Vienna University of Technology (TU Wien) to make a simple Support Vector Regression, measuring the result's accuracy with the Root Mean Square Error.
The results are depicted in figure 13:

Fig.13: Results from estimation of LCOE by a Support Vector Machine

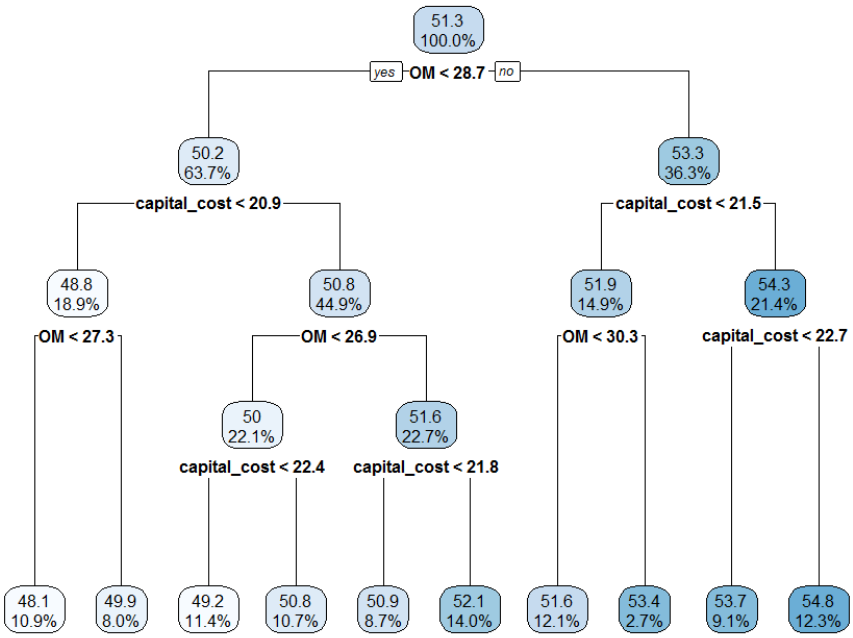


The RMSE of linear model is 23.50403, those of SVM is 1.740835.

Regression Trees

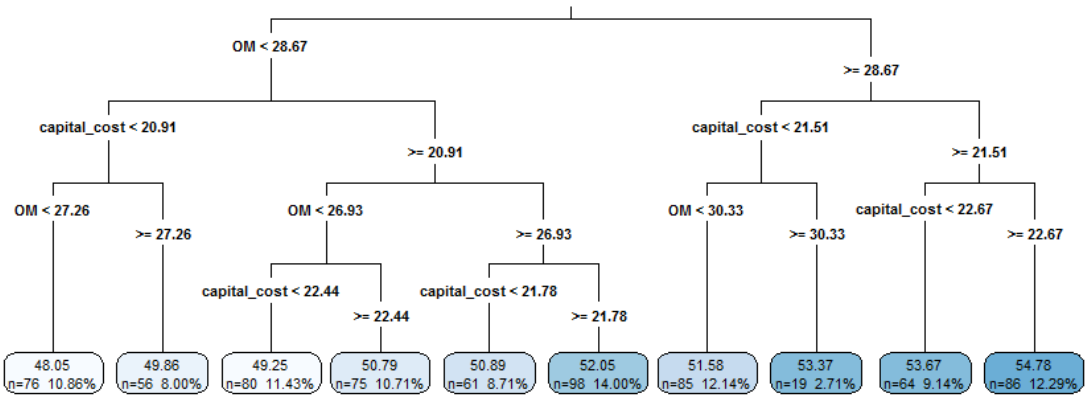
The resulting tree diagram is as follows:

Fig.14: Results from estimation of LCOE by a Regression Trees



In addition to the digits parameter that controls the number of numeric digits to include in the diagram, many other aspects of the visualization can be adjusted. The following command shows just a few of the useful options. The fallen.leaves parameter forces the leaf nodes to be aligned at the bottom of the plot, while the type and extra parameters affect the way the decisions and nodes are labeled:

Fig.15: Results from estimation of LCOE by a Regression Trees



The correlation between the predicted and actual quality values provides a simple way to gauge the model's performance. Recall that the cor() function can be used to measure the relationship between two equal-length vectors. We'll use this to compare how well the predicted values correspond to the true values and the measured value is 0.787181.

A correlation of circa 0.8 is quite good. However, the correlation only measures how strongly the predictions are related to the true value; it is not a measure of how far off the predictions were from the true values.

Another way to think about the model's performance is to consider how far, on average, its prediction was from the true value, we can use the mean absolute error (MAE) that is 1.139048.

This implies that, on average, the difference between our model's predictions and the true quality score was about 1.14. On a simulated LCOE scale from 48 to 55, this seems to suggest that our model is doing relatively good.

On the other hand, recall that most LCOE's are (since we have performed the simulation under symmetric variance assumption) almost uniformly distributed in each class. Therefore, a classifier that did nothing but predict the mean value may still do relatively bad according to this metric.

Indeed, using mean absolute error between actual values and mean value MAE became 1.868438.

Model trees

The used model trees are the M5' algorithm (M5-prime) by Wang and Witten, which is an enhancement of the original M5 model tree algorithm proposed by Quinlan in 1992.To improve the performance of our learner, let's try to build a model tree. Recall that a model tree improves on regression trees by replacing the leaf nodes with regression models. This often results in more accurate results than regression trees, which use only a single value for prediction at the leaf nodes.

The correlation is very high (0.8497821): the improving in MAE make it substantially perfect (0.9962982).

Measuring forecasting Accuracy - Part II

Let's get the whole thing a bit more complicated. In the first simulation, we've used a symmetric variance for all the LCOE parameters of the IEA model data. Now we'll do the same imposing a certain trends for OM, (for the replies from 1 to 3300 the OM vary in a range from 8 USD/MWh to 18 USD/MWh; from 3300 to 6600 the OM vary in a range from 14 USD/MWh to 20 USD/MWh, for 6600 to 10000 the OM vary in a range from 16 USD/MWh to 20 USD/MWh) and to capital cost (for the replies from 1 to 3300 the OM vary in a range from 16 USD/MW to 23 USD/MW; from 3300 to 6600 the OM vary in a range from 22 USD/MW to 26 USD/MW, for 6600 to 10000 the OM vary in a range from 26 USD/MW to 39 USD/MW) for all the other parameters of LCOE equation, a variance equal to zero was settled.

The resulting LCOE is shown in figure 16:

Fig.16: Results from estimation of LCOE under the hypothesis of second case study

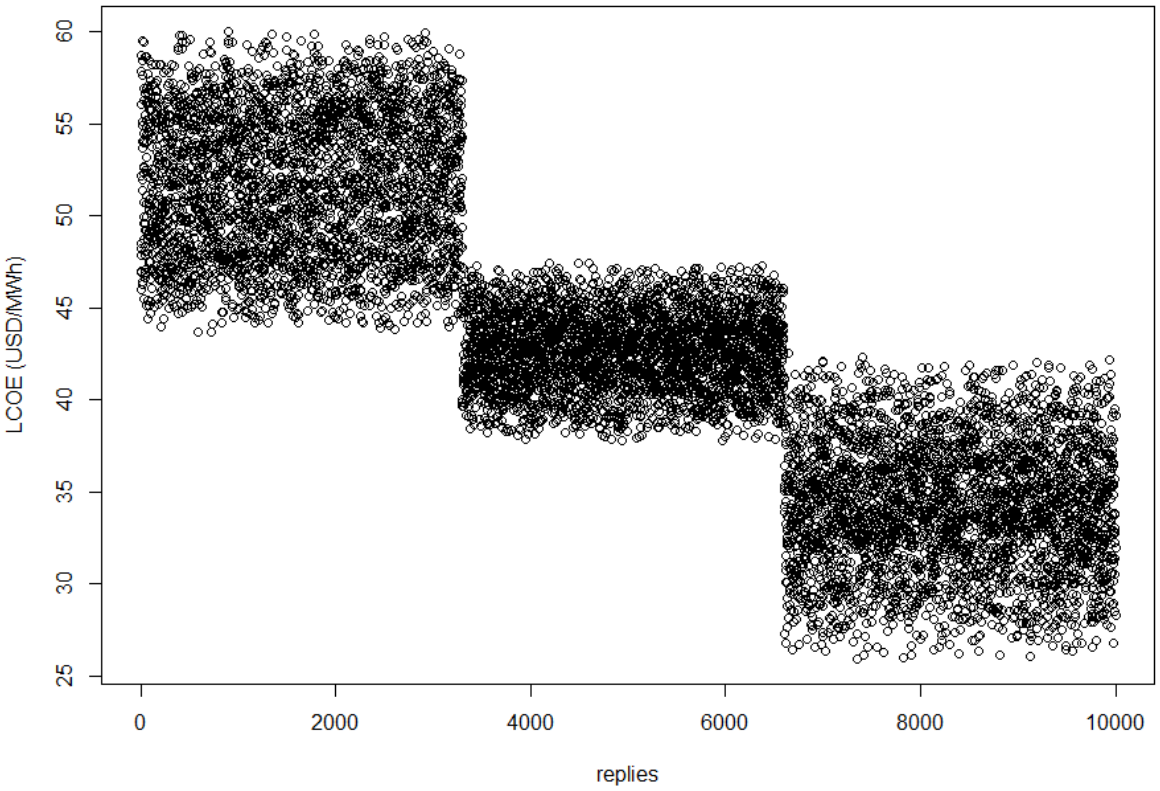
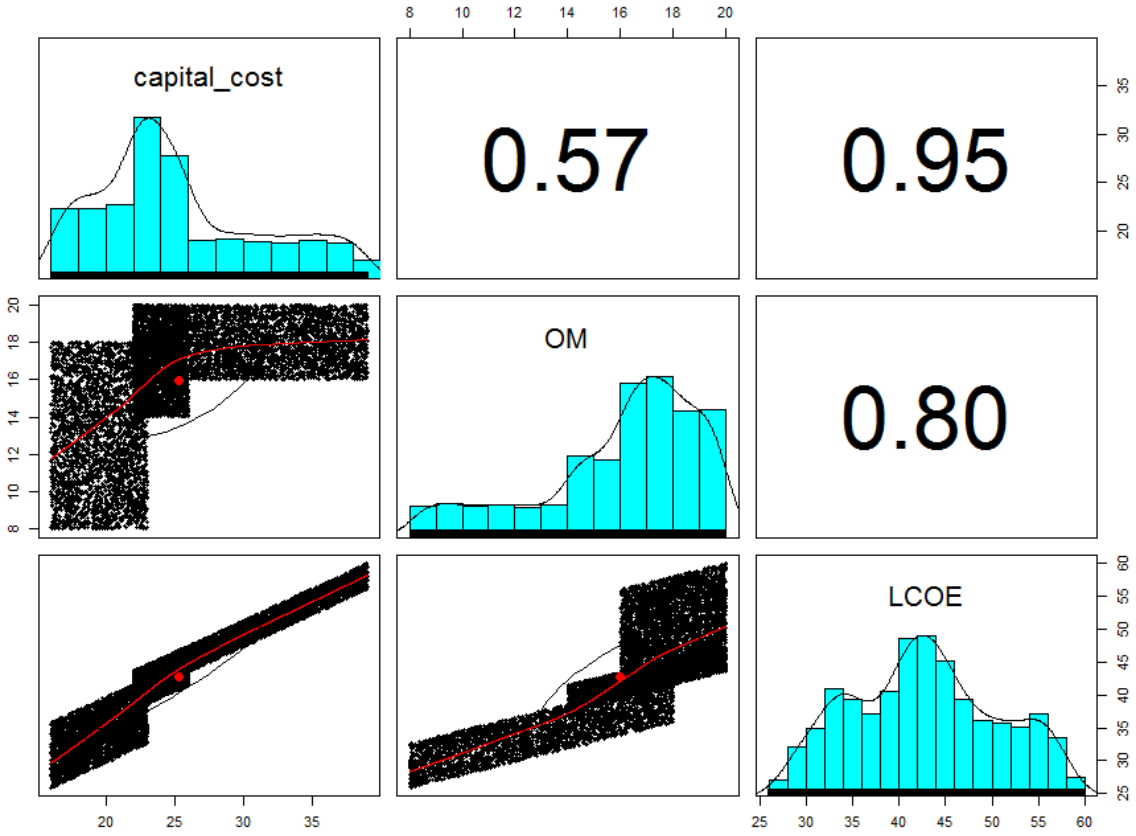


Figure 16 – Profile of LCOE in the case study (cost vs replies)

Linear model

Now, look to the same previous exercise in this changed framework.

Fig.17: Results from estimation of LCOE by a linear model



Using only Operating cost, we get a Multiple R-squared of 0.6416 (Adjusted R-squared 0.6415) with a p-value of < 2.2e-16.

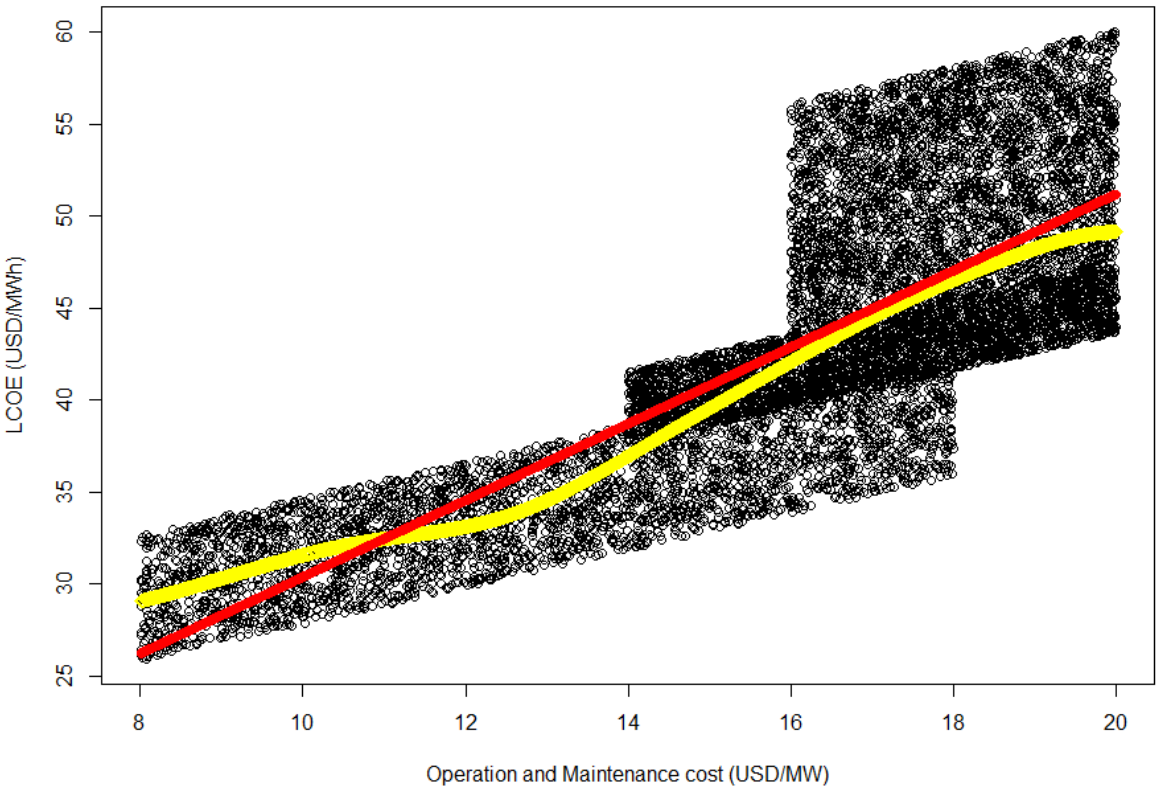
Artificial Neural Network - The Multi Layer Perceptron

Using as a predictor only OM, we get a correlation of 0.8117 with a R^2 of 0.6588.

Support Vector Machines

In this case, the difference became more clear respect to the previous case:

Fig.18: Results from estimation of LCOE by a Support Vector Machine



The RMSE of linear model is 27.010 vs the RMSE of the SVM that is 4.66.

Regression Trees

Fig.19: Results from estimation of LCOE by a Regression Trees

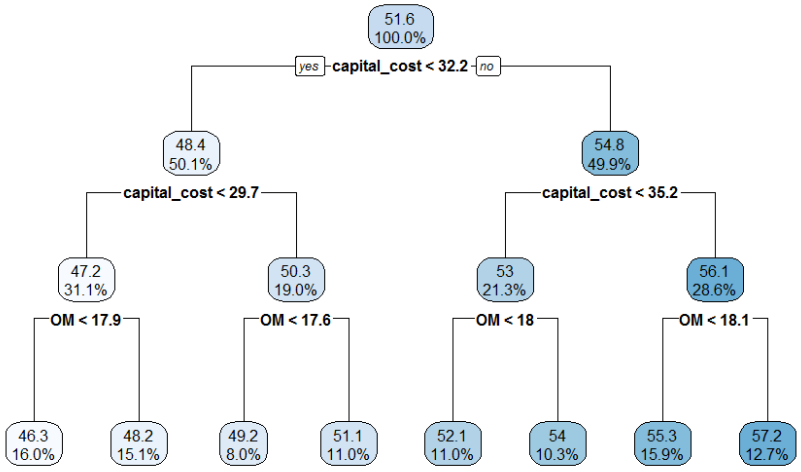
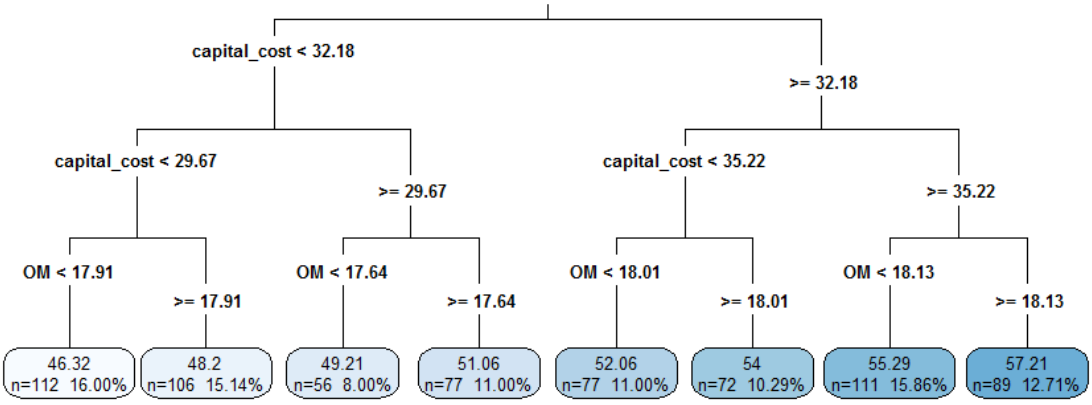


Fig.20: Results from estimation of LCOE by a Regression Trees



The correlation is 0.9539122, the MAE is 0.9000009, a fairly good performance. Using mean absolute error between actual values and mean values (51.58714) MAE became 3.229067, for the same reasons explained in the previous test.

Model Trees

The measured correlation is 0.9967285, the MAE is 0.2368006.

Conclusions

This work represents just a first simple step of an in-depth analysis about the application of several machine learning approaches to energy cost calculations and, more generally, to every analysis of interest on energy system.

The work, based on methodology framework provided by IEA for LCOE, can be applied as well for other type of energy cost and evaluation methods. An example of that could be a methodology for imputation of missing data in energy survey, whether a method to estimate the energy habits of a population.

The applications of machine learning to the energy system are often of an engineering nature [**57-64**]. The proposed work aims to highlight how there is a vast space to use ML also in relation to economic analyzes and in supporting decisions typical of policy making, with simple but significant examples.
In the recent years, the use of ML in energy analysis also about economics aspects it has undergone a strong acceleration in several application [**65-72**]. The scope of this work was limited to a single task, that is the increase in accuracy deriving from models based on ML techniques compared to a traditional one.

Powerful but relatively simplified models were used:

- the neural networks used was those of the simplest conception (the so-called "plain vanilla"), in the simplest possible version, a single layer consisting of a single neuron;
- support vector machines did not benefit from any form of tuning;
- regression trees and regression models were not subjected to any treatment to improve performance.

The results obtained, in some cases far superior to those obtainable from a linear model (SVM), are far enough to demonstrate the superiority of these approaches over inferential statistics and to clearly suggest how fruitful is the direction of investigation they indicate in the execution of this tasks, one of the most significant in energy analysis.

References

1. https://www.tensorflow.org/resources/learn-ml
2. Chollet, Francois - Deep Learning with Python, Manning Publications Co., New York, 2018
3. available at https://unica.it/static/resources/cms/documents/DeepLearningWithPython.pdf
4. IEA (2020), *Projected Costs of Generating Electricity 2020*, IEA, Paris https://www.iea.org/reports/projected-costs-of-generating-electricity-2020
5. Rao, Marco e Tricoli, Carlo, Indicatori di costo nella generazione energetica - Analisi critica del concetto di LCOE: RT/2015/10/ENEA
6. *Ulrich Nissen* and *Nathanael Harfst*, (2019), Shortcomings of the traditional "levelized cost of energy" [LCOE] for the determination of grid parity, *Energy*, **171**, (C), 1009-1016
7. Walter Short, Daniel J. Packey, and Thomas Holt - A Manual for the Economic Evaluation of Energy Efficiency and Renewable Energy Technology March 1995 - NREL/TP-462-5173
8. Aldersey-Williams, John and Rubert, Tim (2019) Levelised cost of energy - a theoretical justification and critical assessment. Energy Policy, 124. pp. 169-179. ISSN 0301-4215 ,http://dx.doi.org/10.1016/j.enpol.2018.10.004
9. Fu, Ran, David Feldman, and Robert Margolis. 2018. U.S. Solar Photovoltaic System Cost Benchmark: Q1 2018. Golden, CO: National Renewable Energy Laboratory. NREL/TP-6A20-72399. https://www.nrel.gov/docs/fy19osti/72399.pdf.
10. Bruck, Maira. A Levelized Cost of Energy Model for Wind Farms That Includes Power Purchase Agreements (PPAS). Diss. University of Maryland, College Park, 2018.
11. Tran, Thomas TD, and Amanda D. Smith. "Incorporating performance-based global sensitivity and uncertainty analysis into LCOE calculations for emerging renewable energy technologies." *Applied energy* 216 (2018): 157-171.
12. Berguglia, C. S., & Vaio, F. (2016). Complessità e modelli. Torino: Bollati Boringhieri.
13. Qin, T. (2020). Machine Learning Basics. In *Dual Learning* (pp. 11-23). Springer, Singapore.
14. Rebala, Gopinath, Ajay Ravi, and Sanjay Churiwala. "Machine learning definition and basics." *An Introduction to Machine Learning*. Springer, Cham, 2019. 1-17.
15. Cohen, Stanley. "The basics of machine learning: strategies and techniques." *Artificial intelligence and deep learning in pathology*. Elsevier, 2021. 13-40.
16. Mehrotra, Dheeraj. *Basics of Artificial Intelligence & Machine Learning*. Notion Press, 2019.
17. Alpaydin, Ethem. *Machine learning: the new AI*. MIT press, 2016.
18. Hopfield, John J. "Artificial neural networks." *IEEE Circuits and Devices Magazine* 4.5 (1988): 3-10.
19. Abraham, Ajith. "Artificial neural networks." *Handbook of measuring system design* (2005).
20. Hassoun, Mohamad H. *Fundamentals of artificial neural networks*. MIT press, 1995.

21. Tealab, A. (2018). Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, *3*(2), 334-340.
22. Zador, Anthony M. "A critique of pure learning and what artificial neural networks can learn from animal brains." *Nature communications* 10.1 (2019): 1-7.
23. Chung, SueYeon, and L. F. Abbott. "Neural population geometry: An approach for understanding biological and artificial neural networks." *Current opinion in neurobiology* 70 (2021): 137-144.
24. Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." *Machine learning* 20.3 (1995): 273-297.
25. Müller, K. R., Smola, A. J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., & Vapnik, V. (1997, October). Predicting time series with support vector machines. In *International conference on artificial neural networks* (pp. 999-1004). Springer, Berlin, Heidelberg.
26. Chapelle, Olivier, and Vladimir Vapnik. "Model selection for support vector machines." *Advances in neural information processing systems* 12 (1999).
27. Weston, Jason. "Support vector machine." *Tutorial, http://www. cs. columbia. edu/~ kathy/cs4701/documents/jason_svm_tutorial. pdf, accessed* 10.0 (2014): 0-5.
28. Kecman, Vojislav. *Support vector machines basics*. School of Engineering, University of Auckland, 2004.
29. Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges, and trends. *Neurocomputing*, *408*, 189-215.
30. Kingsford, Carl, and Steven L. Salzberg. "What are decision trees?." *Nature biotechnology* 26.9 (2008): 1011-1013.
31. Rokach, Lior, and Oded Maimon. "Decision trees." *Data mining and knowledge discovery handbook*. Springer, Boston, MA, 2005. 165-192.
32. Moret, Bernard ME. "Decision trees and diagrams." *ACM Computing Surveys (CSUR)* 14.4 (1982): 593-623.
33. Quinlan, J. Ross. "Decision trees and decision-making." *IEEE Transactions on Systems, Man, and Cybernetics* 20.2 (1990): 339-346.
34. Brodley, Carla E., and Paul E. Utgoff. "Multivariate decision trees." *Machine learning* 19.1 (1995): 45-77.
35. Yang, Yongxin, Irene Garcia Morillo, and Timothy M. Hospedales. "Deep neural decision trees." *arXiv preprint arXiv:1806.06988* (2018).
36. Hu, Xiyang, Cynthia Rudin, and Margo Seltzer. "Optimal sparse decision trees." *Advances in Neural Information Processing Systems* 32 (2019).
37. Turing, A. M. (1950). Computing machinery and intelligence. *Mind, 59*(236), 433–460.
38. Newell, Allen, J. Clifford Shaw, and Herbert A. Simon. "The processes of creative thinking." *Contemporary Approaches to Creative Thinking, 1958, University of Colorado, CO, US; This paper was presented at the aforementioned symposium.*. Atherton Press, 1962.
39. Weizenbaum, Joseph. "ELIZA—a computer program for the study of natural language communication between man and machine." *Communications of the ACM* 9.1 (1966): 36-45.
40. Shannon, Claude E. "XXII. Programming a computer for playing chess." *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 41.314 (1950): 256-275.

41. Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
42. Hopfield, J. J. (1982) Neural networks and physical systems with emergent collective computational properties. Proc. Nat. Acad. Sci. (USA) 79, 2554-2558.
43. Hopfield, J. J. (1984) Neurons with graded response have collective computational properties like those of two-sate neurons. Proc. Nat. Acad. Sci. (USA) 81, 3088-3092.
44. Feigenbaum, Edward A. "Some challenges and grand challenges for computational intelligence." *Journal of the ACM (JACM)* 50.1 (2003): 32-40.
45. Smith, William AP, and Edwin R. Hancock. "Recovering facial shape using a statistical model of surface normal direction." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.12 (2006): 1914-1930.
46. Anyoha, R. *Can Machines Think?* 2017. Accessed 24 November 2019. https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/
47. Cai, Li, and Yangyong Zhu. "The challenges of data quality and data quality assessment in the big data era." *Data science journal* 14 (2015).
48. Peters, Anicia, Michael Oren, and Nicola Bidwell. "Namibian and American cultural orientations toward Facebook." *CHI'12 Extended Abstracts on Human Factors in Computing Systems*. 2012. 2603-2608.
49. Pillai, P., Bolong, J., Osman, M. N., & Hashim, N. (2021). Students' Emotional Engagement in Lecturers' Personal Facebook Account. *Sciences*, *11*(15), 123-133.
50. Costa-Duarte, M. V., Sampedro, L., Molino, A., Xavier, H. S., Herpich, F. R., Chies-Santos, A. L., ... & de Souza, R. C. (2019). The S-PLUS: a star/galaxy classification based on a Machine Learning approach. *arXiv preprint arXiv:1909.08626*.
51. THE MASTER ALGORITHM: How the Quest for the Ultimate Learning Machine Will Remake Our World by Pedro Domingos. New York: Basic Books, 2015. 311 pages, index. Hardcover; $29.99. ISBN: 9780465065707
52. Lantz, Brett. *Machine learning with R: expert techniques for predictive modeling*. Packt publishing ltd, 2019.
53. Koehrsen, Will. "Overfitting vs. underfitting: A complete example." *Towards Data Science* (2018).
54. Domenico, Piccolo. "Statistica - Il Mulino - Bologna." (1998).
55. Trafalis, Theodore B., and Huseyin Ince. "Support vector machine for regression and applications to financial forecasting." *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*. Vol. 6. IEEE, 2000.
56. Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and regression trees. Belmont, CA: Wadsworth. *International Group*, *432*(151-166), 9.
57. Kazem, Hussein A., Jabar H. Yousif, and Miqdam T. Chaichan. "Modeling of daily solar energy system prediction using support vector machine for Oman." *International Journal of Applied Engineering Research* 11.20 (2016): 10166-10172.
58. Eseye, Abinet Tesfaye, et al. "Machine learning based integrated feature selection approach for improved electricity demand forecasting in decentralized energy systems." *IEEE Access* 7 (2019): 91463-91475.
59. Perera, A. T. D., et al. "Machine learning methods to assist energy system optimization." *Applied Energy* 243 (2019): 191-205.
60. Zhang, Liang, et al. "A review of machine learning in building load prediction." *Applied Energy* 285 (2021): 116452.

61. Gao, Tianhan, and Wei Lu. "Machine learning toward advanced energy storage devices and systems." *Iscience* 24.1 (2021): 101936.
62. Shivam, Kumar, Jong-Chyuan Tzou, and Shang-Chen Wu. "A multi-objective predictive energy management strategy for residential grid-connected PV-battery hybrid systems based on machine learning technique." *Energy Conversion and Management* 237 (2021): 114103.
63. Masoumi, Masoud. "Wave Characteristics Prediction Using Raw Data and Multi-Output Machine Learning Algorithms: Towards a Data-Driven Wave Energy System Development."
64. Wang, Zeyu, et al. "Practical issues in implementing machine-learning models for building energy efficiency: Moving beyond obstacles." Renewable and Sustainable Energy Reviews 143 (2021): 110929.
65. Ghoddusi, Hamed, Germán G. Creamer, and Nima Rafizadeh. "Machine learning in energy economics and finance: A review." *Energy Economics* 81 (2019): 709-727.
66. Donti, Priya L., and J. Zico Kolter. "Machine learning for sustainable energy systems." *Annual Review of Environment and Resources* 46.1 (2021).
67. Duchesne, Laurine, Efthymios Karangelos, and Louis Wehenkel. "Recent developments in machine learning for energy systems reliability management." *Proceedings of the IEEE* 108.9 (2020): 1656-1676.
68. Jamil, Faisal, et al. "Peer-to-peer energy trading mechanism based on blockchain and machine learning for sustainable electrical power supply in smart grid." *IEEE Access* 9 (2021): 39193-39217.
69. Magazzino, Cosimo, Marco Mele, and Giovanna Morelli. "The relationship between renewable energy and economic growth in a time of Covid-19: a machine learning experiment on the Brazilian economy." *Sustainability* 13.3 (2021): 1285.
70. Khayyam, Hamid, et al. "Improving energy efficiency of carbon fiber manufacturing through waste heat recovery: A circular economy approach with machine learning." *Energy* 225 (2021): 120113.
71. Cogoljević, Dušan, et al. "A machine learning approach for predicting the relationship between energy resources and economic development." *Physica A: Statistical Mechanics and its Applications* 495 (2018): 211-214.
72. Babenko, Vitalina, et al. "Classical machine learning methods in economics research: Macro and micro level example." *WSEAS Transactions on Business and Economics* 18 (2021): 209-217.

ENEA
Servizio Promozione e Comunicazione
www.enea.it

Stampa: Laboratorio Tecnografico ENEA - C.R. Frascati
ottobre 2022