UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Ingegneria Industriale

Corso di Laurea Magistrale in Ingegneria Chimica e dei Processi Industriali

Tesi di Laurea Magistrale in Ingegneria Chimica e dei Processi Industriali

Machine Learning modelling of die filling for pharmaceutical powders

Relatori
prof. Andrea Claudio Santomaso
prof. Chuan-Yu (Charley) Wu

Laureando:
Stefano Peschiutta

Anno Accademico 2019-2020

Who gets scandalised is always ordinary:
yet, I add, he is also always bad informed.

Pier Paolo Pasolini

Summary

In tablet manufacturing there are still unclear relationships between the equipment specifications, API properties and the deposited mass in the dies. Knowing in advance the filling performances is an important boost for the process optimisation of this industrial application. In order to do so, one linear die filling data-set and one rotary die filling data-set are supplied to some Machine Learning techniques to find out if these specific operations are predictable. For linear die filling three grades of micro-crystalline cellulose (MCC PH-101, MCC PH-102 and MCC DG) are analysed. The best model is found to be a simple Regression Tree and the most significant input feature to determine the deposited mass is the shoe speed. Moreover, for rotary die filling MCC PH-101, MCC PH-102 and MCC CP-102 are considered as materials and Random Forest results in being the best Machine Learning predictive model while the material class is identified as the most significant input feature with the turret speed. Accuracy scores are satisfactory enough for both learning problems, even though some advanced models should be applied to the rotary die filling issue to improve its predictability. From this study, it is clear that the material flowing properties, the shoe speed and the turret speed are the most important and significant features in determining the final product deposited mass and, as a consequence, the goodness of the applied model accuracy.

Contents

List of Figures

List of Tables

Introduction

Preliminary background

Tabletting is one of the most widely used but challenging pharmaceutical processes involving solid materials, specifically, granular materials. Pharmaceutical powders are usually made of active ingredients and inactive ingredients. Active ingredients are biologically active and they are the main drug component; while inactive ingredients, usually called excipients, are the remaining tablet components where active ingredients are dispersed.
Moreover, tablets are the most common oral dosage form (ODF) for drugs to be delivered to the public and they can be sold and manufactured in different sizes and shapes (e.g. spherical, elliptical, compressed, etc.). Every single tablet must contain the same amount of both active ingredients and excpients in order to be reliably considered as a unit of measure for drugs intake, this is a pivotal point for achieving high-quality pharmaceutical products.
Mainly, drug powders are deposited in a die before compression and this step is carried out with different techniques. Nonetheless, die filling is probably the most critical stage during the tabletting process since here the powder flow behaviour controls the final composition, mass and segregation [1]. In this framework, predicting the final composition and mass inside the die is a big issue in the industry.
In order to carry out some predictions, many empirical laws have been introduced based both on powder properties and filling machinery specifications. Despite that, up to today we are still far from understanding the real physical laws that properly link the material properties and the equipment specifications to the final product specifications.
In this framework, *Machine Learning* models seem to be useful and novel tools to try to predict the particle behaviour during the filling step in the tabletting

industry. In fact, these computational methods attempt to find information and patterns from already existing experimental data sets, thus creating a predictive model to be applied to future data related to similar problems.

Objectives

This thesis aims to use Machine Learning models to analyse their predictive performance on the filling process of three different types of *micro-crystalline cellulose* (MCC) and identify the best model as a predictive model for future analyses. Both rotary die filling and linear die filling are studied for the same three different MCC powders. In this project, the sole analysed die filling output is the total mass deposited inside the die.
Moreover, this thesis wants to determine which variables, amongst the given input features, are more relevant and dominant for the final output record thanks to a feature analysis performed with Machine Learning algorithms as well.
Eventually, this study also longs for understanding and explaining how each variable contributes and affects the final output value from learning broad experimental data sets, where physical correlations between the granules properties, equipment specifications and the final deposited mass are not easy to determine.

Thesis layout

This study has the following structure:

- **Chapter 1 - Literature review**

  In this first part, the study recalls and presents the main background about particle size and shape, particle density and powder flowability. In addition to that, the industry of tabletting and die filling is here presented. This theoretical background helps us understand better the whole problem and also can give hints on the final physical explanations of die filling events.

- **Chapter 2 - Machine Learning techniques**

  A brief description of Machine Learning and its rationale is given and all the applied algorithms are described with their advantages and drawbacks.

- **Chapter 3 - Materials and methods**

  Data-sets and analysed types of MCC are presented, also the experimental set up is commented in order to understand how data have been retrieved. Furthermore, the commercial software for Machine Learning modelling are introduced and, finally, the considered performance indices are defined.

- **Chapter 4 - Linear die filling: results and discussion**

  In this chapter a thorough investigation was performed to define which model performs best for the linear die filling data set. A feature analysis is also carried out with the best model and some physical comments are made on the data pattern interpretation.

- **Chapter 5 - Rotary die filling: results and discussion**

  Same as Chapter 4 but the focus was placed on rotary die filling.

- **Chapter 6 - Conclusions**

  Some final comments are provided to summarise the whole study after the previous evidences.

Chapter 1

Literature review

Granular or particle technology is a specific branch of science which aims to describe and outline the physical events that deal with solid and granular materials. More specifically, particle technology is a term used to refer to the handling and processing of particles and powders. Today, this wide term can include also the study of multiphase solid-liquid systems behaviour applied to the industry equipment [2, Introduction].
Nowadays, particles and powders are largely applied in many process industry fields, such as the pharmaceutical industry, the food industry, the metallurgical and plastic industry. Despite that, this very demanded science seems to be underrated in the Chemical Engineering studies and still many issues related to this technology do not have a proper answer. In fact, the difficulty of powder characterisation has led to specific application-oriented equations, preventing from having a general picture of the matter.
Very interestingly, powders are very difficult to model and to picture also because they show some behavioral characteristics of the three phases, solid, liquid and gas. For instance, powders can be compressed like gas until a certain extent; like liquids they can move and have an own fluid dynamics; like solid they can bear some elastic deformation.
Hence, particle technology becomes a new attractive branch of Chemical Engineering, where novel and important breakthroughs can be achieved due to the past lack of industrial interest and of proper predictive tools and knowledge.
Within this context, also *Machine Learning* can play a major active role in learning and predicting some particle science applications, which the theory appears to

be still unclear.

1.1 Powder properties

Granular materials are usually made of two-phase system, one solid phase and one gas-air phase, which impacts on the solid material properties. First of all, granular materials are characterised by their main single granule properties, namely the particle size, shape and density. After that, it is advisable to give details also on powder flowability, which describes the tendency of the particles to flow and to move under an applied stress.

1.1.1 Particle size

For the sake of simplicity, it would be very easy and straightforward to describe and characterise every type of granular material as a heap of spherical-like particles with the same diameter. Unfortunately, every single particle has a different irregular shape and size even if the heap is made of the same material and it comes from the same industrial production line.
Usually, real particles require more than one dimension to completely describe them, and their typical size and shape stand in a range, so it is not acceptable to characterise completely a granular material with a single number, usually we describe a distribution of physical properties [2, Chapter 1].
In fact, a powder material can be described at two different levels, a single particle characterisation level and a bulk material level.
Firstly, particle size is a primary material property which defines and affects secondary properties such as the flowability and granules cohesiveness. On a single particle level, a diameter is used to define the particles size. This diameter can either be an *equivalent diameter* or a *statistical diameter*.
The *equivalent diameter* is associated with a given reference shape and size of an equivalent ideal granule that shares the same given physical property. For example, the *equivalent spherical volume diameter* (x_v)is the diameter of a sphere that shares the same volume of the analysed granule, according to Equation (1.1).

$$x_v = \sqrt[3]{\frac{6}{\pi} \cdot V_{granule}} \tag{1.1}$$

With the same procedure, we can define other equivalent diameters according to different properties such as the equivalent surface diameter, the equivalent specific area diameter or the equivalent specific surface diameter just giving a reference geometry. A specific focus is given to the equivalent projection circle (EQPC), which is the diameter of a circle that has the same area of the projected area of the particle, according to Eq. (1.2). In order to carry our these calculations we must know in advance the real properties of each granule to match their equivalent size. In order to do so we need proper analytical experiments in the granular material.

$$EQPC = \sqrt{\frac{4 \cdot A_{projected,granule}}{\pi}} \tag{1.2}$$

Another approach to determine the single granules diameter involves the use of a *statistical diameter*. The *statistical diameter* is usually associated with image analysis and optical microscopy performed on a sample of the granular material. If we take a digital snapshot of a granule, we consider a 2D projection of its shape and dimension. In addition to that, we must define a reference direction along which we measure the particle statistical size. A very common example is the Feret's diameter (x_F), which is the distance between two given tangents on opposite sides of the particle. According to the reference tangent, x_F can vary a lot for each particle and it typically has a maximum and a minimum possible values. The process for retrieving the Feret's diameter is illustrated in figure 1.1. Similarly, we can define the Martin's diameter (x_M) as the chord that perfectly bisects the particle projected area, and we can also define the Shear diameter (x_{Sh}) as the minimum translation length of the particle along a given direction.
Moreover, size is represented by a distribution function in the bulk level. Every single particle property is a distributed property in the bulk level since every granule contributes to define the bulk powder material. As a matter of fact, we can built histograms either based on number frequency ($f_N(x)$) or mass frequency ($f_M(x)$) of the selected single particle size range with respect to the total considered number or mass of particles. The considered size can be one of the defined equivalent or statistical diameter. This way, a bell-shape distribution is

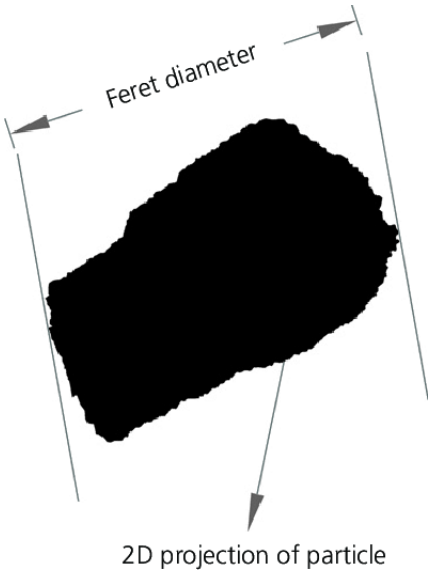


Figure 1.1: Feret's diameter (x_F) illustration from a digital image.

obtained along the reference particle size, this is called a *particle size distribution* (PSD).
Every PSD can roughly be approximated by a *probability density function* (PDF). Typically, the most commonly used PDF are the normal distribution function (1.3), which is symmetrical, and the log-normal distribution function (1.4), which is skewed.

$$f_i(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \tag{1.3}$$

$$f_i(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln(x)-\mu)^2}{2\sigma^2}\right) \tag{1.4}$$

Every single distribution has some typical statistical *moments* (μ_k), which normally describe the behaviour of the distribution. The definition of a statistical moment is given at Eq. (1.5), where x is the given particle size. The first moment ($\mu_1 = \mu$) is defined as the mean of the distribution, while the second moment ($\mu_2 = \sigma^2$) is named variance. The square root of the variance is the standard deviation of the distribution (σ). The definition of the used PDF for the analysed PSD, the definition of the mean and of the standard deviation is typically enough to describe completely the behaviour of the bulk size of the examined

granular material. It is worth mentioning that also the mode and the median of the distribution can be reported as PSD descriptors.

$$\mu_k = \int_0^\infty x^k f_i(x)\, dx \tag{1.5}$$

Furthermore, the *cumulative distribution function* (CDF) is the cumulative value of the considered PDF from the minimum size value to the maximum. Analytically, it is simply the integral of the PDF in its domain and it values between 0 and 1. We usually refer to CFD with the symbol $F_i(x)$, according to Eq.(1.6). It can either refer to a mass or number cumulative function. Every CDF can be represented as an undersize cumulative function or as an oversize function, simply considering $1 - F_i(x)$. In Figure 1.2 and example of two different PSD is shown with a log-normal distribution function.

$$F_i(x) = \int_{x_{min}}^{x_{max}} f_i(x)\, dx \tag{1.6}$$

Every particle size distribution can be effectively charted by the related PDF or CDF with their appropriate statistical moments. So that, we are able to utterly describe the bulk material characteristic size. All the theory related to this section is found in [3],[2, Chapter 1],[4], [5, Chapter 2] and [6].

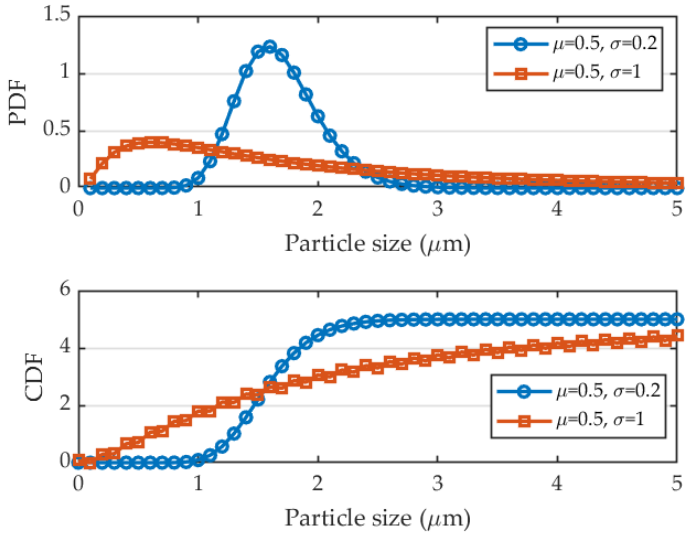


Figure 1.2: Example of two log-normal PSD with their relative PDF and CDF for a granular material.

1.1.2 Particle shape

Particle shape is a very complex property to measure and another primary properties of granular material that affects the secondary properties such as flowability. In fact, the rougher the granule and the less the material is able to move due to internal friction. Such as particle size, the shape can be determined both on a single material level and on a bulk perspective. Particle shape can differ a lot in the same heap of material, strongly depending also on the industrial production method.
On a single particle level, the first approach is the definition of some standard ratios between the granule dimension, better known as Heywood ratios. For instance, the *elongation* is the ratio between the particle length and width and the *flakiness* is the ratio between the width and the thickness [5, Chapter 2]. However, these ratio are not really straightforward since very different granule shape can entail the same Heywood ratio.
In order to solve this problem, one simply way to quantify shape from optical microscopy is represented by shape factors. The most important shape factors are Wadell's sphericity (Ψ) and circularity (Φ_c), where [3]:

$$\Psi = \left(\frac{\text{surface area of a sphere}}{\text{surface area of the particle}} \right) \Bigg|_{\text{same Volume}} \tag{1.7}$$

$$\Phi_c = \left(\frac{\text{circumference of a circle}}{\text{perimeter of the projected particle}} \right) \Bigg|_{\text{same proj. Area}} \tag{1.8}$$

The closer to 1 Ψ and Φ_c are and the more spherical and rounded the examined granule is [5, Chapter 2]. Therefore, after some simple laboratory analyses, a good shape predictor is obtained.
Some more sophisticated and accurate shape predictors are the *shape coefficients* and the *fractal dimension*. This last method aims to replicate and draw the granule perimeter as a sum of touching yardstick with a given equal length. The smaller the yardstick length and the more accurate is the granule perimeter description.
Eventually, on a bulk level of interest, the powder shape is again defined as a distributed property such as the powder size. Hence, a PDF, a CDF and statistical descriptors are required to fully characterise the bulk shape of the granular material.

1.1.3 Particle density

Particle density is a crucial primary property. Density is formally a ratio between the mass and the volume. The mass of the granular material is easily found out just by measuring it with a well-calibrated balance; despite that, volume is not so easily quantified in granular materials. In fact, every powder contains a certain amount of air that contributes to define the material volume. Air can be trapped between different granules but also inside each single granules, determining inner voids.
Hence, we can define different particle densities according to the particle volume definition. Moreover, bulk density for a granular material has also a different definition based on how the material is handled, this defines the amount of air between each granules.
Firstly, on a single particle level, the *true* density (ρ_t) is defined considering the volume of the granules excluding open and open pores and is the density of the solid material like it would not have any type of void [6]. *True* density is measured with liquid or gas pycnometry, the laboratory procedures are illustrated in [7], [3] and [6].
Furthermore, the *skeletal* or *apparent* density is defined considering the skeletal volume of the granules, so excluding only open pores with the external surface and including close pores.

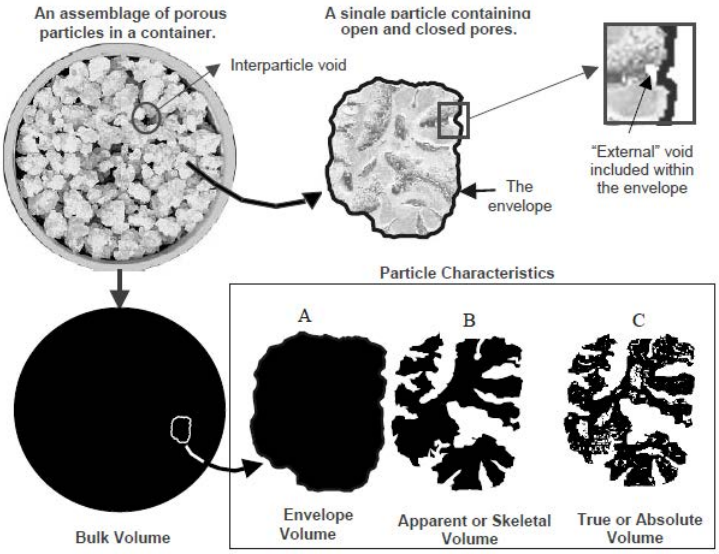


Figure 1.3: Framework of possible volumes in a granular material [8].

At last, *envelope* density refers to a particle volume that includes both open and close pores. In Figure 1.3 an illustration of all possible granule volumes is provided.
On a bulk level, powder bulk density (ρ_b) depends on its consolidation and packing state, the more the powder is packed and the higher the density because of a smaller bulk volume. Therefore, there are several definition of ρ_b based on the material application and process. Surely, if the *true* density and the *bulk* density are known, powder porosity (ε) is easily found with Eq. (1.9):

$$\varepsilon = 1 - \frac{\rho_b}{\rho_t} \tag{1.9}$$

Eventually, the following *bulk* densities are defined [4],[9]:

- **Tapped density**, bulk density of a powder which has been compacted by tapping or vibration following a given standard. It can be carried out with a fix mass or volume procedure.
- **Poured density**, it is the bulk density of a poured powder into a cylinder thanks to a funnel. We refer to it thanks to the standard ISO 3923-1.
- **Dispersed density**, the powder is poured into a fix volume from an higher distance and it passes through a sieve that aerates it and disperses it. Usually, very cohesive powders are very difficult to disperse. This method was developed by Santomaso et al. and details in [10].

1.1.4 Flow properties

Granular materials are able to show a flowing motion after the application of a normal stress. More specifically, a granular material is able to transport a shear stress after the application of a normal stress. Powder flowability is not an intrinsic property of materials, but it changes in time due to time consolidation, humidity and particle-air fluidisation. In addition to that, flowability is strongly influenced by the acting forces between single particles, namely capillary and electrostatic forces due to surface tension and surface net charges [11]. These types of interactions can define the material cohesiveness and its ability to flow freely under a given stress.
Particle flow properties started being studied in the middle of the last century thanks to hoppers applications. Actually, particle storage and discharge in hoppers used to be a very alluring event to study because of its industrial application. In this operation, particles undergo the gravity force activity both during the discharging and storing steps. Thus, it is crucial to understand the flow properties of the examined material in order to predict whether it moves or not while the aperture is open.
Even though powder flowability is not an easy property to determine, there are several methods that aim to characterise this feature. *Indirect* methods use particle properties, supposed to be correlated to flowability, in order to define some flow indices such as the *Angle of repose*, the *Hausner ratio* or the *Packing ratio* [10]. However, these methods are not trustworthy for every material, they must be applied in some reliability ranges.
Moreover, *direct* methods are more commonly used and universally accepted to define the flowability and the cohesiveness of a granular material. These methods are based on direct observations of the material during its flow and in different consolidation states. One of the most renowned *direct* method is the determination of the *flow factor* (ff) with a Jenike Shear Cell [12].
This procedure requires a ring shear cell, illustrated in Figure 1.4, where the analysed material is loaded inside. At the beginning, a pre-shear step is applied. In this step a normal pre-shear stress (σ_{pre}), or *consolidation stress*, is applied and the cell starts moving radially until a steady state is reached and the measured shear stress becomes constant. After that, a normal stress (σ) is applied on the cell lid, it is lower than σ_{pre}. Then, the cell starts applying a time-increasing shear stress

(τ) to the material until it yields and an *incipient flow* is shown. When the material yields, the shear stress is recorded and the material is brought again to the original consolidation state with another pre-shear step. After that, every time a different lower consolidation normal stress is applied and its relative yielding shear stress is recorded. This way, it is possible to obtain an *internal yield locus* (IYL) curve for the analysed material, that can be usually approximated to a linear behaviour described by Coulomb's Law (1.10). At the pre-shear condition the powder flowability should not depend anymore on its previous mechanics history. For this reason we must bring back to the original steady state the material before another yielding procedure. The pre-shear point is the upper bound of the IYL [13].

$$\tau = \mu\sigma + c \tag{1.10}$$

In Coulomb's Law, μ is the *friction coefficient* and c is the is called *cohesion coefficient*. The *internal yield locus* curve represents the point at which the material fails under a given consolidation state. Hence, Mohr's circle can be used in the IYC curve to determine the material *unconfined yield stress* (f_c) and the *major principal consolidation stress* (σ_c). More specifically, σ_c is identified by the highest values of the Mohr's circle tangent to the upper bound of IYL and passing through the x-axis. It represents the critical failure condition of the material in the considered pre-shear condition. Moreover, f_c is the stress identified by the circle tangent to IYL and passing through the origin of the axes. It represents the major shear stress bore by an arch of the unconfined material in the given consolidation state σ_{pre} [2, Chapter 10].

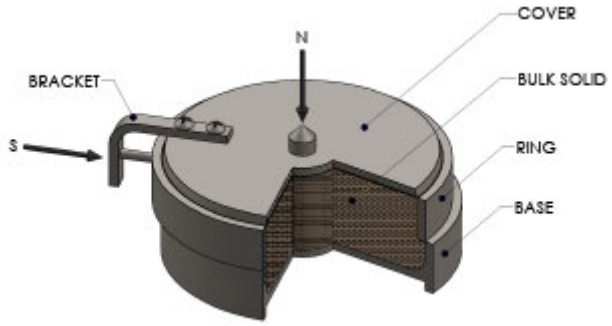


Figure 1.4: Jenike Shear Cell picture.

If this procedure is repeated several times for different pre-shear stress, many IYL curves are obtained, thus a lot of representative couple (σ_c, f_c) are gathered for different consolidation state, namely different bulk density for the material. Thanks to that, we are able to understand which is the unconfined yield stress that an arch of free material can bear under a given stress, which is also a description of powder flowability. In fact, for a given consolidation stress, the lower f_c and the more flowing the material. Many couples (σ_c, f_c) define the *flow function* of the material, Figure 1.5, and the average of the ratio (1.11) is called *flow factor*.

$$ff = \frac{\sigma_c}{f_c} \tag{1.11}$$

In order to establish the cohesiveness of the studied material, some typical values of the *flow factor* have been proposed to characterise the powder flowability. If ff is lower than 2, the material is usually non flowing, if included in between 2 and 4 the material is cohesive, in between 4-10 it is fairly-free flowing, while over 10 the powder is usually free-flowing [9].

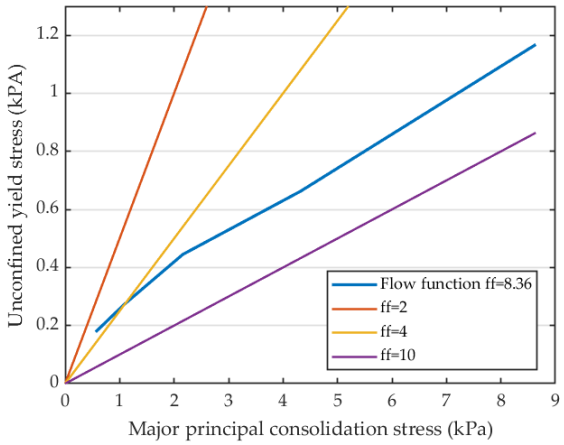


Figure 1.5: Example of the determination of the flow factor from a flow function for maltitole from a course assignment.

Nevertheless, for our study it is worth mentioning also other type of flowability indices that are less commonly used. One is the *flow index*, that is the smallest orifice of a steel cylinder that enables the powder to flow. It is expected that very cohesive powders have a bigger orifice than free flowing powders. In addition to that, we can consider also the average mass flow rate of discharged material

through an aperture as a very empirical method to interpret the particle flowability.

1.2 Tablet manufacturing

Tablet manufacturing is a commonly used production process in the pharmaceutical industry. It aims to measure out an API (active pharmaceutical ingredient) powder blend and to compact it in a pill or tablet in order to provide simple and straightforward drug dosage for customers. A normal API pharmaceutical blend contains one or more API, based on the desired effect of the drug, and some excipients that can be used as stabilisers, preservatives, diluting agents or binders. Glidants and lubricants excipients are very important in terms of industry production since they ease the material filling into the die and the tablet ejection die after the compression [11].
The tabletting manufacturing process is mainly divided in three steps: die filling, compression and ejection. The first two are crucial in controlling the final product quality [14]. In fact, during these steps many segregation phenomena can arise, thus producing an heterogeneous final product. It is essential to achieve repeatable and adequate die filling to ensure the product quality. Die filling is usually carried out either by a feed hopper or shoe passing linearly over dies, or by a rotary die filling equipment made of a circular die table that moves over a stationary shoe.
Die filling can be due to the gravity force that allows the powder to fall into the die from the feeder; or by suction thanks to a piston inside the die that sucks the material inside the mold.
After this step, the considered blends is usually compressed by an upper punch and particles are subjected to high stresses, which lead to both an elastic and plastic deformation of the material and, eventually, also a mechanical crushing. After the punch release, the material is able to recover the elastic energy but the overall volume has been reduced and the material has been compacted into a tablet or a pill [15],[11].
A brief description of the tablet manufacturing steps is provided in Figure 1.6.
Granular material flowability is a key feature to analyse and rate the goodness of the die filling step, since it is expected that a very cohesive material flows poorly into the die and the final discharged mass of the material is more affected

also by the velocity of the shoe. In fact, different type of feeder velocities induces different consolidation states inside the hopper where the granular material is located.

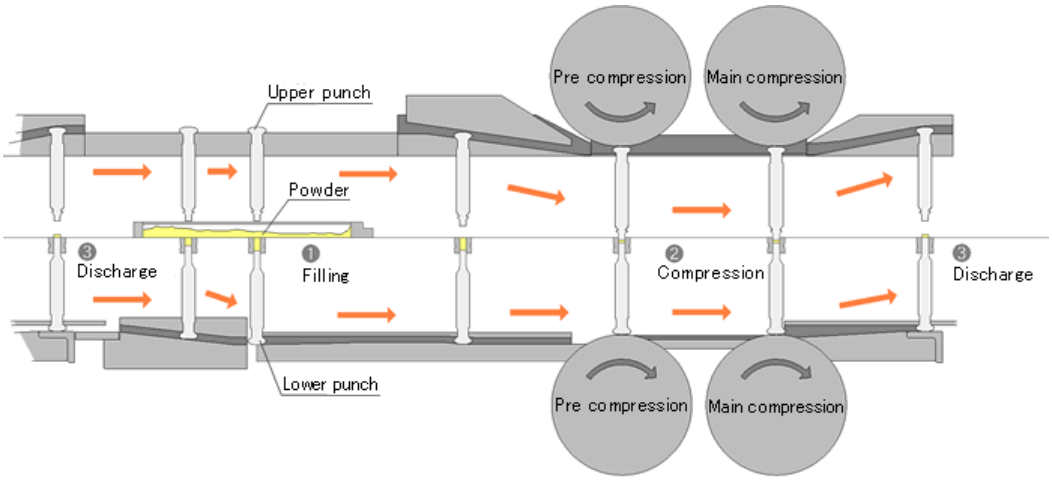


Figure 1.6: Tablet manufacturing scheme.

1.2.1 Linear die filling

In this study, the analysed pharmaceutical powders undergo two different types of die filling. The first one is the probably the most commonly used: gravity induced linear die filling.
In this simple way of filling, a powder is left to fall into a static die by only the gravity force from a shoe that passes over it with a proper velocity, named *filling velocity or shoe speed* (v_s). A simple sketch of the processes is shown in Figure 1.7. In this specific type of filling, particle flowability and the machinery specifications play a key role on the characterisation of the flow behaviour during the discharging step and in terms of filling efficiency. For this kind of problem, the filling efficiency is defined by the *fill ratio* (δ), defined in Equation (1.12) as the ratio of the discharged mass into the die and the maximum mass in a fully filled die [1],[17].

$$\delta = \frac{m}{M} \tag{1.12}$$

In addition to that, it is possible to define a critical velocity of the shoe v_c over which the die can be only partially filled. In fact, it is reasonable to think that the

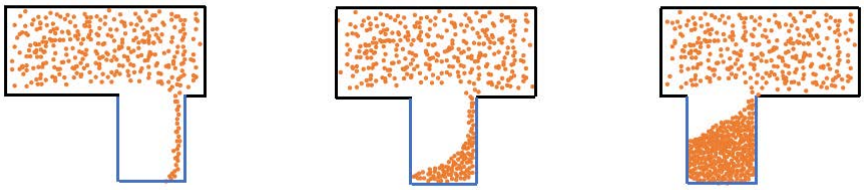


Figure 1.7: Illustration of linear die filling process driven by gravity: initial state; intermediate state; final state [16].

lower the shoe speed, the closer to 1 the fill ratio. Hence,δ can be expressed as follows for shoe velocity v_s bigger than v_c [14]:

$$\delta = \left(\frac{v_c}{v_s}\right)^n \tag{1.13}$$

where n is a material-dependant parameter between 0.9 and 1.6 for common tested powders.

1.2.2 Rotary die filling

The other considered filling method is the rotary die filling. Although linear die filling is vastly investigated, this type of filling that simulates real industrial applications is not still well understood. The equipment is made of a circular table, named turret, that has multiple dies on its edge. The turret spins clockwise over an hopper that contains the considered powder. Contrary to the previous example, here the feed frame is fixed and dies are moving under the material. In the specific application, the feed frame has a rotary paddle that spins anti-clockwise to distribute the material over the moving dies. This set-up really mimics the real industrial equipment for API tabletting and offers the possibility to characterise the filling problem from another perspective.
Also for this problem, the fill ratio (1.12) is taken as an index for the filling efficiency. In Figure 1.8 an illustration of the applied set-up is available.
Here, we refer to the critical filling velocity as the lowest velocity of the turret over which we can only have partially filled dies. This analysis is particularly

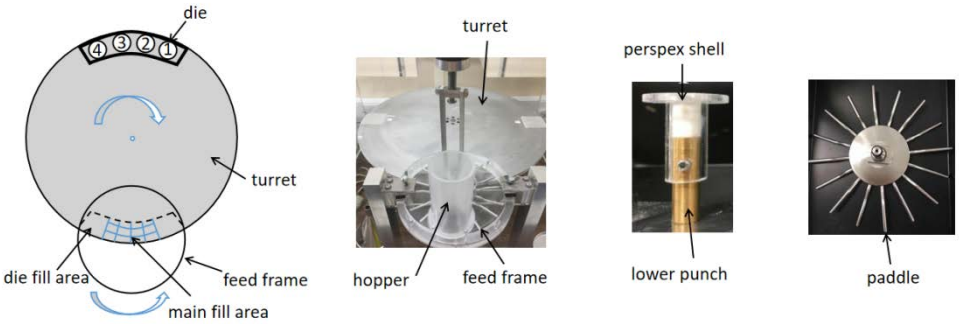


Figure 1.8: Upper view, front view, the die, the paddle. Rotary die filling system scheme [16].

challenging because we consider the filling of multiple subsequent dies in the same time, differently from classic studies on die filling where a single die is considered.
For rotary die filling, the material properties, the turret speed and the paddle speed can all affect the filling process and the flow behaviour [16]. Nonetheless, it is still not clear how all these features interacts with each other in determining the final discharged mass in the die. Therefore, a statistical analysis and some Machine Learning techniques can be applied to this filling process in order to find some hidden relationships between these variables and the final fill ratio.

Chapter 2

Machine Learning techniques

Artificial Intelligence (AI) is an advanced data science method that allows machines and computers to show a kind of intelligence like the one of living beings and humans. With AI, machines are able to show they can gain knowledge, understand problems, adapt their approach to the surrounding environment, give instructions and gain new skills [18, Introduction]. These type of characteristics make the AI very similar the human intelligence in many applications. Thanks to that, in the latest years many robots have been developed to substitute and aid the man labour in the heavy industry and in the electronic field. Furthermore, in the recent years, *androids*, robots with human appearances that use AI, seem to be in the spotlight as a new prominent human facility.
Nevertheless, if AI involves problems recognition, drawbacks evaluation, problem solving, robot control and a sufficient result prediction, how do machines *learn* all these features?
In fact, intelligence is real only after a learning step. Thus, also machines must recognise patterns and rules in order to become more *intelligent* and solve simple tasks. Therefore, Machine Learning (ML) is a sub-field of AI that creates codes, computer architectures and algorithms in order to make machines *learn* from their previous experience and from existing database [18, Introduction].
Typically, learning problems are input-output problems where the machine has to find the correlation between variables, which can be either continuous numbers or categorical labels or nouns. For instance, imagine that a function f exist between the input vector $\mathbf{X} = (x_1, x_2, \ldots, x_n)$ and the output vector $\mathbf{Y} = (y_1, y_2, \ldots, y_n)$ and the learner has to guess it. In order to fulfil this exercise, all

the renowned Machine Learning techniques split the input $\mathbf{X}$ in a training set and a testing set. The learner tries to guess the correlations within the training set and it usually finds a function h, named *function estimator*, that aims to be as similar as possible to f. Then, the machine uses the function h to predicts the testing set, thus having $h(\mathbf{X})$ as output according to Eq. (2.1).
After this step, prediction errors (ε) are evaluated to characterise the prediction quality and wellness [19, Chapter 2].

$$\mathbf{Y} = f(\mathbf{X}) = h(\mathbf{X}) + \varepsilon \tag{2.1}$$

Each different Machine Learning algorithm has its own internal parameters values $\mathbf{w}$ which define the learning components of the AI learning model. By minimising the root mean square error (RMSE, Eq.(2.2)) between the real output of the training set $\mathbf{Y}$ and the predicted one with the *function estimator* $h(\mathbf{X})$, the ML model finds the optimal value of $\mathbf{w}$, therefore it also recognises the best features pattern [20]. The definition of RMSE is the following:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - h(x_i))^2} \tag{2.2}$$

where N is the total samples in the training set. This accuracy scorer is useful only if there is a *regression* problem involving only numerical learning. When the machine needs to learn categorical values, this is no longer applicable and we are in front of a *classification* task. In classification issues, AI must no longer find numerical correlations between variables but it has to recognise and categorise inputs into different classes. Here, ML algorithms are completely different from the regression one, they are mainly based on decision making rather than numerical fitting. However, also in these learning problems, the initial data-set is split into a testing and a training sub-sets [19, Chapter 2].
Furthermore, in terms of preliminary dictionary, we can refer to *supervised learning* when the ML model receives a set of categorical or numerical inputs and outputs and it has to approximate the input to the known output in both regression or classification problems. Whereas, we refer to *unsupervised learning* when we simply have input terms and not output terms to take as a reference for our approximation. Unsupervised learning models are definitely the most difficult type of algorithms to create and implement for Machine Learning purposes; in

fact, here we guess and try to explore patterns and characteristics rather than numerical relationships. [18, Introduction].
This study only handles supervised regression learning problems since all the provided data-sets involve only numerical values and the fill ratio of the die is known as the sole output of the system.

To sum up, in regression supervised learning problems we split the initial data-set into a training (or learning) set ($\mathcal{L}$) , where the Machine Learning algorithm is applied to create a correlation model between input and output feature, and a testing set ($\mathcal{T}$), where the new model is applied to predict the output the remaining real input data and an accuracy analysis is carried out.
Therefore, the way of splitting and resampling the training and the testing sets is a crucial step to have a precise and accurate model. Usually, resampling approaches can be computationally demanding, because they involve fitting the same ML method several times using different $\mathcal{L}$ subsets. The most commonly used resampling and data-splitting method are the *K-fold cross-validation* and *bootstrap* methods
Every data-splitting method gives a slightly different model to feed the computational intelligence predictive machine, so it is a decisive step to take while setting up the initial part of the applied ML algorithm. Specifically, this study is about to use only *K-fold cross validation* method for partitioning the data-sets.
This method randomly divide the initial provided data in evenly-sized K non overlapping folds, or sub-sets. Then, it removes one fold and applies the ML algorithms to the *K-1* groups that forms the training sets $\mathcal{L}$. The removed sub-set is used ad the testing set $\mathcal{T}$. After that, the fitting procedure is carried out for the *K-1* groups and a RMSE is computed as an accuracy scorer. This procedure is repeated K times; every time a different fold is used as a training set and a new RMSE is produced. Finally, the final accuracy scorer is the average of the K repetitions, as shown Eq. 2.3.

$$\overline{\text{RMSE}} = \frac{1}{K}\sum_{i=1}^{K}\text{RMSE}_i \tag{2.3}$$

This equation can be applied to every different type of accuracy scorer. It is clear that the number of folds affect a lot the final prediction accuracy of the model, K can ideally span from 2 to the data-set size. However, the bigger K and the

more expensive the computational demand. For the sake of simplicity, and as a standard procedure, K=10 is taken in this analysis.
All theory related to this paragraphs is found in [19, Chapter 5],[18, Chapter 5].

2.1 Artificial Neural Network (ANN)

One of the employed Machine Learning techniques for supervised learning is the Artificial Neural Networks (ANNs) algorithm. This model is probably one of the most commonly used and popular algorithm to learn a regression rule.
This technique fits the input-output response with a network of non-linear elements interconnected with calibrated weights. Here, each single network element is called *neuron* [18, Chapter 10], [21, Chapter 5].
This is due to the fact that the model structure reminds to a biological neural network where an input is processed and analysed through different *neurons* before seeing the final outcome. In fact, originally ANNs were designed to mimic the brain activity [18, Chapter 10].
The are many available ANNs structures, but the most popular is the *Multilayer Perceptron* stucture (MLP). In this case, the network is divided intro three different parts: an input layer, an hidden layer and on output layer. Each layer has a different number of neurons; the input layer includes as many nodes as the number of different inputs, the output layer has as many neurons ad the number of real outputs, while the hidden layer has a variable number of neurons that defines a specification parameter in the algorithm. Each neuron receives the signal from every node present in the previous layer and sends the reassessed signal to every neurons of the following layer, as charted in Figure 2.1 [20].
In a MLP structure, only the hidden layer nodes and the output layer nodes are computational nodes. Each computational neuron in the hidden layer works quite simply. The n-dimensional input vector $\mathbf{X}$ is summed and weighted with a n-dimensional weight vector $\mathbf{W}$ using the dot product $\mathbf{X} \bullet \mathbf{W}$, that is represented by the following equation:

$$U = \mathbf{X} \bullet \mathbf{W} = \sum_{i=1}^{n} x_i w_i \tag{2.4}$$

Typically, inputs features are normalised in the domain [-1;1], thus the dot products is still included in this domain if all weights are smaller than 1. Positive

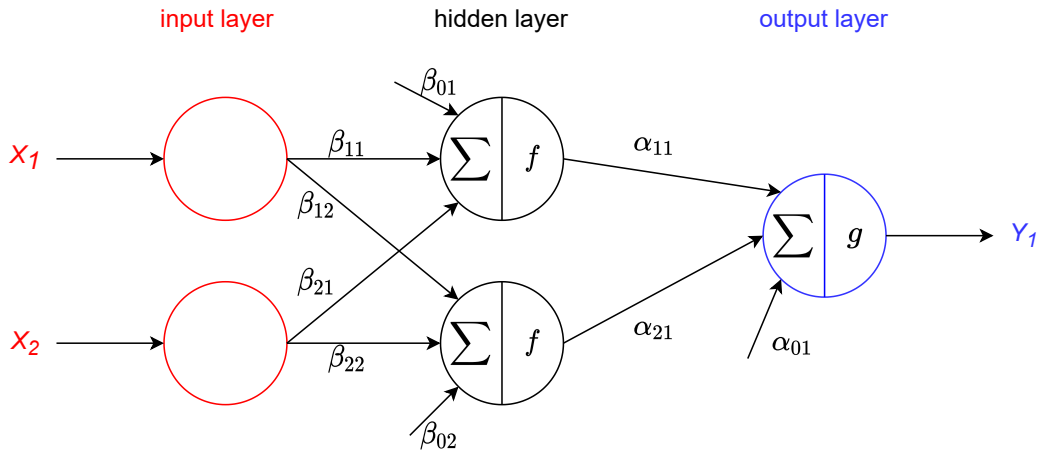


Figure 2.1: Multilayer perceptron structure for a 2 inputs, 1 output system and with 2 neurons in the hidden layer. f and g are the activation function and $\beta_{01}, \beta_{02}, \alpha_{01}$ are the weights.

weights identify an excitatory synapses, while negative weights represents inhibitory synapses [18, Chapter 5]. The value of U is then compared to a threshold weight θ, or β_{0i}, α_{0i} in Fig. 2.1, given for each node. If U$\geqslant \theta$, then the output is 1 and the signal is passed to the following computational burden; on the contrary, if U$\leqslant \theta$, the signal is stopped and the output is 0. This logical sequence of comparing a value to a threshold is named *threshold logic unit* (TLU).
When a signal can successfully pass through the TLU step, it must be processed through a *activation function* (f), which is a normal signal *transfer function*. This function represents the magnitude of the node output for the next layer neurons. Some typical *activation function* are the *sigmoidal function* (2.5), the *gaussian curve* and the *logistic curve*. These functions return always a value in between [-1;1], hence it becomes the new value for the following neuron input.

$$f(\mathbf{W_0} + \mathbf{W} \bullet \mathbf{X}) = f(\mathrm{U}) = \frac{1}{1 + \exp^{-\mathrm{U}}} \tag{2.5}$$

All this node structure is also called *perceptron*. Eventually, when the model is learnt and applied to the testing set, the final output value is de-normalised and compared to the real data. Therefore, a MLP structure learns in the training set how to properly regress a system just by looking at the final output approximation and changing the values of the single weights inside the computational

layers. The model can be more accurate also by adding an increasing number of neurons and hidden layers. This specific network layout and learning procedure is also called a *feedforward neural model*. Besides, there are many way to calibrate and train a MLP network but the most employed model is the *back-propagation error* algorithm.
MLP is one of the strongest ML techniques for supervised regression problems, but sometimes it can incur *overfitting* and it is high computational demanding. *Overfitting* occurs when the applied algorithm is too complicated for the analysed data and it has too many parameters with respect to the learning features size. It results in a large test error.

This discussion related to ANNs models is referenced in [18, Chapter 10],[21, Chapter 5] and [22, Chapter 11].

2.2 Flexible Neural Tree (FNT)

Artificial neural networks have been improved and extended since their introduction in order to be optimised. One of the following ANNs development is represented by *flexible neural trees* (FNTs), which are a specific type of *feedforward models* with a tree-like structure proposed by Chen and al. in [23].
An FNT is composed by three main parts: leaves, branches and nodes. A leaf is simply a terminal node where an input feature is chosen and fed to the the following node, which is a computational node. Contrary to ANNs, FNTs can have more than 3 layers and inputs nodes (leaves) can be introduced in every different layer. Moreover, branches stand for the weighting connections between various twigs [20]. The single computational nodes runs similarly to ANNs nodes, they share the same *perceptron* structure. Anyway, FNTs nodes can refer and work with different number of inputs signal in the same layer, while in ANNs every computational node has the same number of inputs with respect to the same layer. This is due to the fact that a leaf (an input node) can be introduced everywhere in the tree structure. A simple FNT structure is shown in Fig. 2.2.
Ideally, each computational node can employ a different activation function, but the used FNT algorithm is optimised by using a simple *Gaussian* function (2.6):

$$y_i = f_i(a, b, \mathrm{U}) = \exp\left(-\left(\frac{\mathrm{U} - a_i}{b_i}\right)^2\right) \tag{2.6}$$

where U is given by Eq. (2.4), a_i and b_i are activation function parameters. Furthermore, tree structure and the tree parameters are optimised using respectively the *Genetic Programming* (GP) and the *Particle Swarm Optimisation* (PSO) algorithms [20].

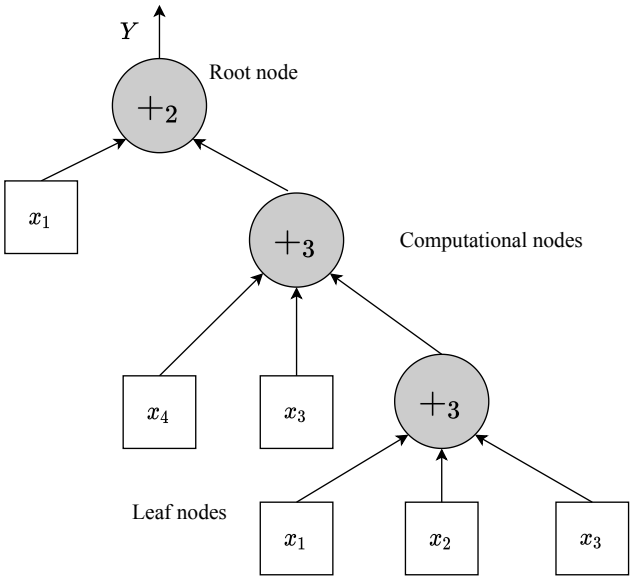


Figure 2.2: Simple sketch of FNT structure. Squares represent leaf nodes, circles represent computational nodes and each arrow is the branch of the tree, namely the weight contribution. Grey colour stands for a Gaussian activation function and $+_i$ for the number of input signals.

GP is population algorithm inspired by the Darwinian evolution theory [21, Chapter 16], it runs endless to find the best tree structure that gives the optimal regression approximation and the minimal tree size. The process is completely probabilistic and try to mimic what happens in a real natural population of living elements. In our case, the elements of the population are many random tree structures. Each elements is initially assigned to a survival probability thanks to a *wheel of fortune* model [21, Chapter 16]. Then, the survivors undergo a mating and crossover procedures along the possible sub-tree structures. In a nutshell, we can describe the applied GP algorithm with the following steps [24], [20]:

- *Selection*. From an initial population P of random tree-structure, we apply

the wheel of fortune to save an offspring population Q of different types of tree structure that fill a *mating pool*. From Q, two *parents* are chosen for the following mating steps.

- *Crossover*. Two different trees are partially swapped and overlapped to substitute sub-tree structures.
- *Mutation*.Terminal nodes can be replaced with new terminal nodes, namely the input leaf is mutated. In addition to that, computational nodes can be replaced with leaf nodes or even we can add a new sub-tree to a randomly chosen computational node.
- *Recombination*. The initial population P and the offspring population Q are combined to return the population R.
- *Elitism*. At last, a number of structures equal to $size(\mathrm{Q})$ is deleted. These must be the worst trees in terms of final approximation results. Eventually, the final tree structure population defines the new population P for the GP cycle. This is repeated until the best tree structure is obtained.

Once the best tree structure is found out in terms of outcome approximation, also the tree parameters can be optimised in order to achieve even an higher model accuracy. There are several bio-inspired and heuristic techniques that have been developed to overcome this problem trying to mimic the social and biological behaviour of physical systems like the Genetic Programming (or *Genetic Algorithm*) does [25].
Many of these techniques have been applied to our study in the FNT software, like the *Grey Wolf Optimisation*, the *Differential Evolution*, the *Artificial Bee Colony* and the *Bacteria Foraging Optimisation* meta-heuristic models. However, PSO is the model that shows the best accuracy prediction, hence it is chosen as the meta-heuristic model to optimise FNT internal parameters.
This technique is one of the best known swarm procedure to improve model parameters and look for an optimal solution. It aims to reproduce the social behaviour of birds in a flock; each set of parameter $(\mathbf{a}, \mathbf{b})$ is considered a single bird (or particle) with a specific position, direction and speed inside the flock population. The total number of possible parameters builds up the entire flock of solutions, which is moving. During the flock motion, each bird can decide to

leave it or to stay with it based on loyalty and selfishness indicators [25],[26]. With this procedure, the position of the i-particle in the flock is defined as:

$$X_{i,d}^{t+1} = X_{i,d}^{t} + V_{i,d}^{t+1} \tag{2.7}$$

where X is the position vector, V is the velocity vector, t is the iteration index and d refers to the space dimension. Moreover, $V_{i,d}^{t+1}$ is determined as:

$$V_{i,d}^{t+1} = w \cdot V_{i,d}^{t} + c_1 \cdot rand_1^t \cdot \left(best_{i,d}^{t} - X_{i,d}^{t} \right) + c_2 \cdot rand_2^t \cdot \left(best_{g,d}^{t} - X_{i,d}^{t} \right) \tag{2.8}$$

where w is the inertia weight factor, $best_{i,d}^{t}$ and $best_{g,d}^{t}$ are the most optimist positions of the p and d individuals inside the swarm along the d dimension at time t; c_1 and c_2 are the loyalty and selfishness parameters whereas $rand_i^t$ are random numbers evenly distributed within the interval [0,1] [25],[26]. Repeating this procedure in loop with many iterations, we can find the optimal values of the FNT parameters. Therefore, combining the tree structure optimisation and the parameter tuning, the final FNT solution is provided. The whole procedure is summed up in Figure 2.3. Using this algorithm a 10-fold cross-validation method was employed to validate the Machine Learning technique. All setting values used in the FNT software are shown in Table 2.1.

Generally speaking, the main limitation of FNT models is their capability of only returning a single-output node, while ANN and MLP models can work with multiple output nodes. Moreover, during the cross validation step, if we apply a k-fold CV the FNT structure is fixed at the beginning and the remaining (k-1) folds optimise only the FNT parameters.

Table 2.1: Parameters values and setting for FNT optimisation procedure.

Parameter name	Purpose	Value
Tree height	The maximum levels of a tree	5
Tree arity	Maximum siblings of a node in a tree	5
Tree node type	Type of activation function	Gaussian
GP population	Total candidates in a GP population	Linear filling: 200 Rotary filling: 400
Mutation probability	Frequency of mutation	0.2
Crossover probability	Frequency of crossover	0.8
PSO population	The initial size of a PSO population	50
Node range	Bound of activation function inputs	[0;1]
Number of folds	Folds of *k*-fold CV	10

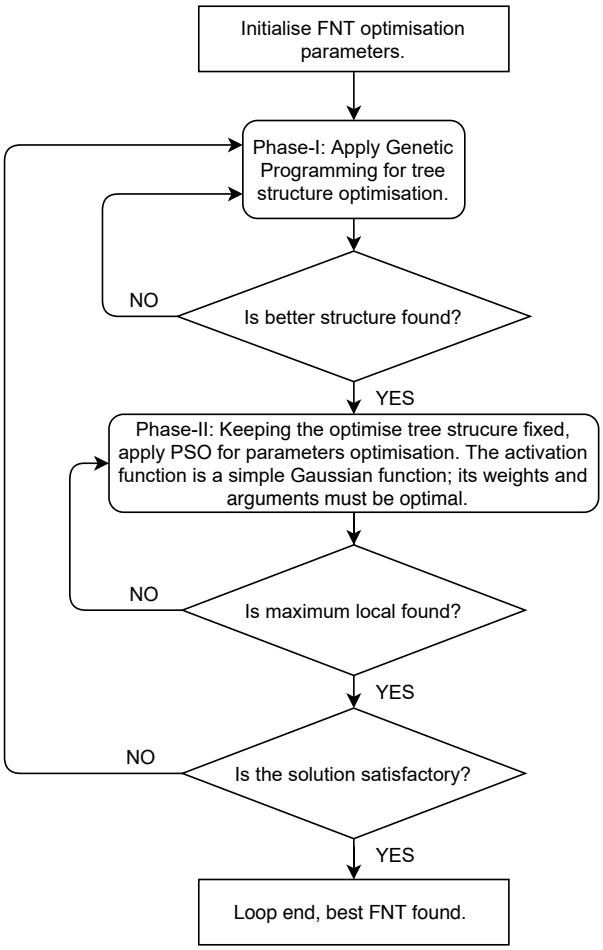


Figure 2.3: Overall scheme of a FNT optimisation technique.

2.3 Regression Tree

Further on, *tree-based* models represent another type of very commonly used Machine Learning technique. Tree models can either carry out classification and regression tasks in a supervised learning problem. For regression purposes, these models imply splitting and stratifying the input space **X** in smaller sub-sets where to perform multiple regressions; this method is generally known as *recursive partitioning regression* [18, Chapter 9].
The splitting strategy is crucial for the wellness of the prediction, it must minimise the total root mean square error performed in the testing set. Basically, the input space is divided into J non-overlapping regions and for every j-region a different regression is carried out [19, Chapter 8]. In our cases, the employed regression model in described by Breiman et al. in [27].
The structure of the final outcome is very similar to the one proposed by FNT in Fig. 2.2. Here, leaves are defined by every single j-regions and can be introduced in every tree level, while the root node is the initial feature from which the splitting strategy starts. With an *upside down* scheme, we can understand the final splitting strategy since every branch is defined from the root node by a numerical interval of the input space. In Figure 2.4 a simple scheme of a Regression Tree is provided. This algorithm allows also to define the number of inner levels of the tree and, consequently, its depth. The deeper the tree, the finer the accuracy but also the higher the tree complexity and computational demand.
Generally speaking, Regression Trees are a regression implementation of Decision Tree models for supervised classification problems.

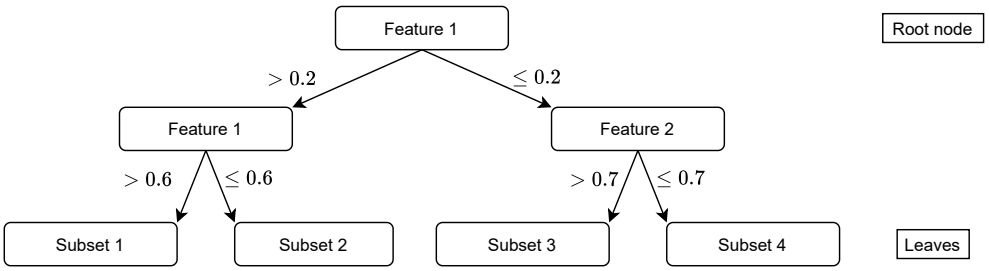


Figure 2.4: Small scheme of Regression Tree defined by 2 input features and 4 subset splits.

2.4 Reduce Error Pruning Tree (REP-Tree)

When we build a tree-based structure, we would like to keep its size as small as possible to improve the model simplicity. Nevertheless, reducing the size usually means also to obtain a worse outcome approximation. Having said that, we would like to analyse how much each branch or leaf affects the final approximation error in order to prune them if their contribution is minor or negligible.
Reduced Error Pruning Tree (REP-Tree) is an computational improvement of Decision Tree models to check out if any internal node or branch can be pruned out to simplify the tree structure and limit the final error increase.
This implementation is a good way to get rid of *overfitting* for simple data-sets. Thanks to that, if *overfitting* occurs, the overall error can be even reduced with an REP-Tree strategy [19, Chapter 9],[20]. The used REP-Tree implementation is based on a Machine Learning software released by WEKA©.

2.5 Random Forests

Random Forests are a framework of another very popular type of Machine Learning models for regression and classification supervised learning: *Ensemble* models. This type of framework is also the last ML technique used for predictive modelling in the examined study. Basically, *Random Forests* were introduced by Breiman in 2001 [28] as an implementation of Ensemble models for Decision Trees. Here, a large collection of independent Decision Trees is built and then they are all averaged. In many issued, Random Forests behave similarly to another Ensemble model: *Boosting*. Thus, it is a very popular and common technique in ML problems since it is also quite simpler to train and tune [22, Chapter 15]. For regression tasks, Random Forests algorithms display the mean of the accuracy indices over all the built Regression Trees. When building these trees, a random split is chosen inside each tree and a random sample of m features is taken from the initial inputs. The tree split must contain only those m predictors while building it. If one of the initial input is a strong and relevant predictor, the models ends up selecting it many times in different trees and causing them to become correlated. A more detailed explanation of this model framework can be found in [22, Chapter 15], [19, Chapter 8] and [29, Chapter 11].

2.6 Feature analysis

Eventually, a feature analysis is carried out on both data-sets to understand and evaluate the importance and the impact of each input variable on the final desired outcome. Very simply, once the best Machine Learning model is found out for every data-set in terms of prediction accuracy, that model is used to evaluate again the wellness of the output prediction accuracy on the reduced data-set. Consecutively, one feature is randomly withdrawn from the data-set and new accuracy indices are computed, then the withdrawn feature that returns the best prediction indices is ultimately discarded from the data-set, reducing its dimension. This is performed again until only 1 feature remains in the data-set which is assumed to be the most important in terms of significance and impact on the predictive modelling problem.

Chapter 3

Materials and methods

In this chapter, the characteristics and the details of the examined data-sets and the Machine Learning tools are presented. All powders properties refer to the study discussed by Zakhvatayeva et al. [30]; while, experimental settings and equipment for rotary die filling are the same used by Zakhvatayeva et al. [16]. Moreover, linear die filling set up is introduced in [14].

3.1 Analysed data-sets

3.1.1 Linear die filling

The linear die filling data-set gathers information about two different grades of micro-crystalline cellulose (MCC): MCC Avicel PH-101, MCC Avicel PH-102 and one co-processed micro-crystalline cellulose and anhydrous dibasic calcium phosphate powder, known as MCC DG. In addition to that, the studied materials are bought from FMC Biopolymer, Cork, Ireland and they have been sieved and characterised in previous studies before introducing them in the filling set up and powders properties are listed in Table 3.1 and in Table 3.2. The data-set is made up of 4 inputs columns and 1 single output. The target output feature is singled out by the mass (g) measured inside the examined die. Each die filling measurement was run in triplicate [20] and then recorded. The actual range for the mass output is [3.75; 45.91] (g).
Moreover, the presented input features are 4: the material true density ρ_{true} (kg/m^3), the material mean diameter d_{50} (µm) ,the granule upper size limit (µm) and the

shoe speed (mm/s) that runs linearly over the die.
There are only three values for granule mean diameter as a feature since they individually refer to the three used materials. The mean diameter (d_{50}) is identified as the size below which 50% of the material is found.
The granule upper size limit is defined as the upper size of the powder segment analysed after a sieving stage. In fact, all materials were sieved into 6 different classes (0-90, 90-250, 250-500, 500-1000, 1000-1400, 1400-2360 µm), and the upper size of each class is taken as an input feature for the analysed powder [20]. Moreover, seven different shoe speeds were used to fill a single die for each mass class of every material. As a result, shoe speed spans between [10; 500] (mm/s).
Eventually, the data-set is built out of 391 rows. In Figure A.1 and Figure A.2 we find the distribution of the used shoe speed and the considered upper size limit value for this data-set divided by the three materials.

Table 3.1: Powders properties involved in linear die filling experiments (data expressed as mean ± standard deviation). The particle size refers to EQPC. $\Psi_{c,50}$ is Wadell's sphericity referred to the first half of the material property distribution. v_c is the critical velocity for linear die filling. [20] and [30]

Material	ρ_{true} (kg/m^3)	d_{50} (µm)	$\Psi_{c,50}$	v_c (mm/s)
MCC PH-101	1581	59.83	0.65 ± 0.01	28
MCC PH-102	1570.3	94.7	0.71 ± 0.01	50
MCC DG	1785.6	52.33	0.71 ± 0.01	25

Table 3.2: Powders flow properties involved in linear die filling experiments (data expressed as mean ± standard deviation). [30]

Material	Flow Function (ff)	Flow Index (mm)	Mass flow rate through a 24 mm orifice (g/s)
MCC PH-101	7.24 ± 1.96	20	47.5 ± 4.4
MCC PH-102	17.96 ± 4.36	12	45.5 ± 0.8
MCC DG	10.04 ± 3.56	22	60.0 ± 5.0

From the material properties charted in the previous tables, it is possible to identify MCC PH-101 as the statistically smallest powder, while MCC PH-102 has the biggest granules. On the contrary, MCC PH-101 displays the worst rounded

shape while the other two materials tend to be slightly more spherical. In addition to that, MCC PH-102 is listed as the most free flowing material according to all the presented flow properties. However, it is hard to say which is the least free flowing material between MCC PH-101 and MCC DG; for sure their flow properties are quite similar and worse than MCC PH-102 ones. According to the classes presented in Figure 1.5, these two powders can be classified as fairly-free flowing in terms of flow function value [9].
To be thorough, it is possible that some outliers or typos are present in this provided experimental data-set. In fact, rows identified by IDs 184, 185, 279 and 391 show values that seem to either extremely higher or lower than the remaining data-set group of values. These data might affect later the accuracy of the predictive models since the *Machine Learning* tools are not able to discard likely outliers from the analysis.

3.1.2 Rotary die filling

The second part of this study focuses on a rotary die filling data-set. The employed data are provided by some experimental analyses carried out by Tang, Zakhvatayeva et al. [16]. This application considers three different grades of MCC: MCC Avicel PH-101, PH-102 (FMC; Biopolymer, Cork, Ireland) and Mycrocrystalline Cellulose Spheres, known as MCC CP-102 (CELP-HERE® CP-102; Asahi Kasei, Japan). Powder properties are charted in Table 3.3 and Table 3.4.
In this experiment, a circular turret with four dies spins clockwise under a feed frame where the powder is inserted by the means of a hopper. The feed frame has a paddle with 16 blades that spins in opposition to the turret motion and helps the material to fill the dies. The paddle and the turret use different rotors.
Again, the data-set is very simple. It is made out of four input features and one single output. Here the output is represented by the fill ratio (g/g) as defined in Eq. (1.12); it spans from 0.22 to 1.08 g/g. Moreover, the four inputs are: the material ID (1 for MCC PH-101, 2 for MCC PH-102 and 3 for MCC CP-102), the die number (1, 2, 3, 4), the paddle speed (rpm) and the turret speed (mm/s).
There are only 5 employed velocities for the paddle speed for each material (10, 30, 50, 70, 100 rpm) while the turret speed is comprised between [8.36; 735.21452] (mm/s). Different distributions of turret speed are applied to the three powders since they have different critical velocities (v_c Eq. (1.13)) to fill the dies. These

different speed distribution applied by materials are found in Figure A.3. In fact, if a material has a very high critical velocity, an higher turret speed must be applied to detect a partially filled die. Critical velocities are linked to the material properties and, usually, a strongly free flowing material has a very high critical velocity since it is more likely to fall from a small aperture during the rotary motion [31]. The idea of critical velocity introduced by Wu in [32] is suitable for linear filling equipment but it is still not clear if it can work even for rotary die filling. Thus it is desirable to investigate if there are some correlation between powder characteristics and the filling performances.
This provided data-set contains 614 rows and each measurement was taken in triplicate.

Table 3.3: Powders properties involved in rotary die filling experiments (data expressed as mean $\pm$ standard deviation). The particle size refers to EQPC. $\Psi_{c,50}$ is Wadell's sphericity referred to the first half of the material property distribution. v_c is the critical velocity for linear die filling. [30]

Material	ρ_{true} (kg/m^3)	d_{50} (µm)	$\Psi_{c,50}$	v_c (mm/s)
MCC PH-101	1581	88.08 $\pm$ 0.03	0.65 $\pm$ 0.01	28
MCC PH-102	1570	108.62 $\pm$ 3.69	0.71 $\pm$ 0.01	50
MCC CP-102	1600	186.60 $\pm$ 0.9	0.91 $\pm$ 0.01	148.2

Table 3.4: Powders flow properties involved in rotary die filling experiments (data expressed as mean $\pm$ standard deviation). [30]

Material	Flow Function (ff)	Flow Index (mm)	Mass flow rate through a 24 mm orifice (g/s)
MCC PH-101	7.24 $\pm$ 1.96	20	47.5 $\pm$ 4.4
MCC PH-102	17.96 $\pm$ 4.36	12	45.5 $\pm$ 0.8
MCC CP-102	14.44 $\pm$ 1.50	5	64.8 $\pm$ 1.0

From the given tables, it is clear that MCC CP-102 is the most spherical material and coarse material, it resembles a smooth spherical-like heap of granular particles. Furthermore, MCC CP-102 is even the best free flowing analysed powder since it shows the slowest resistance to a gravity motion though an orifice and its flow index is the smallest one. Nevertheless, MCC PH-102 seems to be the best free flowing material in terms of flow function (ff) even if its orifice falling

performances are the worst ones. Also looking at the the critical velocities values for linear die filling, we expect MCC CP-102 to be the most free flowing material, followed by MCC PH-102 and MCC PH-101, even if it might be possible that these last two material could show an inverse response in terms of gravity motion from a small aperture.

3.2 Filling equipment

The linear die filling equipment is depicted by Ojha et al. [20] and by Zhang et al. [33]. It is made of a die of 13 mm in diameter and a shoe moving over it equipped with an hopper as a feed frame. The shoe is driven by a pneumatic driving unit, a positioning controller and a displacement transducer. Moreover, a crafted gravity-fed roll compactor with two counter rotating smooth rolls of a diameter of 200 mm and a width of 46 mm is used to compress the tablets for ribbons productions.

The rotary die filling equipment is described by Zakhvatayeva et al. [16]. It consists of a circular die table mounted on a turret, four dies and a forced feed frame with an hopper. The turret is crafted in aluminium and the pitch radius is 205 mm; the pitch radius is defined as the distance between the centre of the turret and the centre of the dies. The feed frame is 199 mm in diameter and has an inner paddle with 16 cylindrical blades provided by FETTE COMPACTING, Germany. The paddle and the turret are moved by two different rotors, the turret spins clockwise while the paddle always spins anti-clockwise. To further mimic the real industrial application, multiple dies (4) are filled at the same time in a series by the feed frame to highlight a possible mass variance between different dies. Dies have a diameter of 7.8 mm and a depth of 12 mm; they are made in Perspex wall and a copper base. Experiments are conducted both at constant turret speed varying the paddle speed and at constant paddle speed varying the turret speed to investigate their reciprocal influence on the final outcome, namely the fill ratio.

3.3 Machine Learning tools

The Machine Learning techniques presented in Chapter 2 are applied to the considered data-sets by the means of two main Machine Learning software. Artificial Neural Network, Regression Tree, REP-Tree and Random Forest algorithms are applied to the existing learning problem thanks to KNIME 4.0® (www.knime.com), an open source analytical platform where many data manipulation tasks can be carried out. With this free statistical software we can first of all prepare the data for the manipulation, sort it out and clean it from dirty measures or unwanted variables. Then, we can apply some statistical or ML algorithms, visualise the results and export them. In this platform, some simple Machine Learning models are ready to use and some inner parameters can

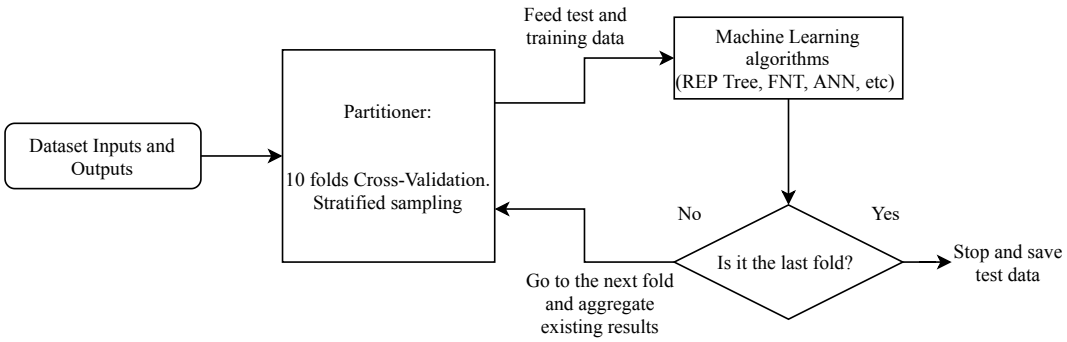


Figure 3.1: Logic block diagram used for predictive modeling with KNIME 4.0®.

In addition to that, Flexible Neural Tree are applied to the considered problems thanks to a self-crafted software written in Java GUI by Ohja. It can be found at dap.vsb.cz/aat/. The detail specifications used in this application are listed in Table 2.1 and PSO is used as an optimisation parameter algorithm for both data-sets.

3.4 Performance indices

In order to evaluate the predictive accuracy of the employed models, some performance indices should be introduced to compare the results of different learning models. One of this, probably the most commonly used one, is the Root Square Mean Error (RMSE) as defined in Eq. (2.2). At every fold loop, a new index is stored based on the testing set model approximation and, eventually, an average index is proposed over the 10 different folds as illustrated in Eq. (2.3). Besides, we propose as an alternative accuracy indices also the *correlation coefficient* Eq. (3.1) and the *determination coefficient* Eq. 3.2, which in our case is simply the square of the *correlation coefficient*.

$$r = \frac{\sum_{i=1}^{N}((h_i - \overline{\mathbf{h}}) \cdot (y_i - \overline{\mathbf{y}}))}{\sqrt{\sum_{i=1}^{N}(h_i - \overline{\mathbf{h}})^2} \cdot \sqrt{\sum_{i=1}^{N}(y_i - \overline{\mathbf{y}})^2}} \tag{3.1}$$

$$R^2 = r^2 \tag{3.2}$$

where h_i is the predicted output, $\overline{\mathbf{h}}$ is the mean of all predicted outputs, y_i is the real output and $\overline{\mathbf{y}}$ is the mean of all real outputs in the testing set. The correlation coefficient (r) spans from -1 to 1 and it is an indicator of the prediction performance. The closer to 1, the best the performance; while the closer to -1, the worst the performance. R^2 is introduced as the square or r to simply have a performance index that spans from 0 to 1. In our study, it can be useful since r is never negative. The best scenario in our case is represented by the fact that all predicted outputs are equal to the real data, this is a displayed by a simple bisector line between the first and the third quadrant of a 2D plane whose axes are labelled by the real data and the predicted data. In this case r should be equal to 1. r is -1 when the models predict exactly the sign-contrary value with respect to the real data. However, since all provided output data are positive, it is very hard that the models predict multiple negative numbers, so we are pretty sure that r is never going to be negative and R^2 can be introduced as an accuracy index. Also in this case, an average of r and R^2 is provided across the 10 folds of the cross validation method on the testing set. Eventually, the standard deviation of r (σ_r) related to its values on all the 10 folds can be added as a supplementary information.

3.4.1 Optimisation loops in KNIME 4.0®

In order to achieve a RMSE as small as possible during the predictive analysis, some optimisation loops are introduced in KNIME. These loops basically control some inner parameters of the various ML models employed during the learning analysis such as the tree levels for the Regression Tree or the number of layers for the Artificial Neural Network. The optimised values of the considered model specifications are reported in Table 3.5.

Table 3.5: Optimised model specification parameters for the algorithms used in KNIME 4.0®.

Data-set	Machine Learning Model	Parameter	Value
Linear Die Filling	Artificial Neural Network	Layers	7
		Neurons per layer	6
	Regression Tree	Tree levels	10
	Random Forest	Tree levels	11
Rotary Die Filling	Artificial Neural Network	Layers	3
		Neurons per layer	7
	Regression Tree	Tree levels	10
	Random Forest	Tree levels	12

Chapter 4

Linear die filling: results and discussion

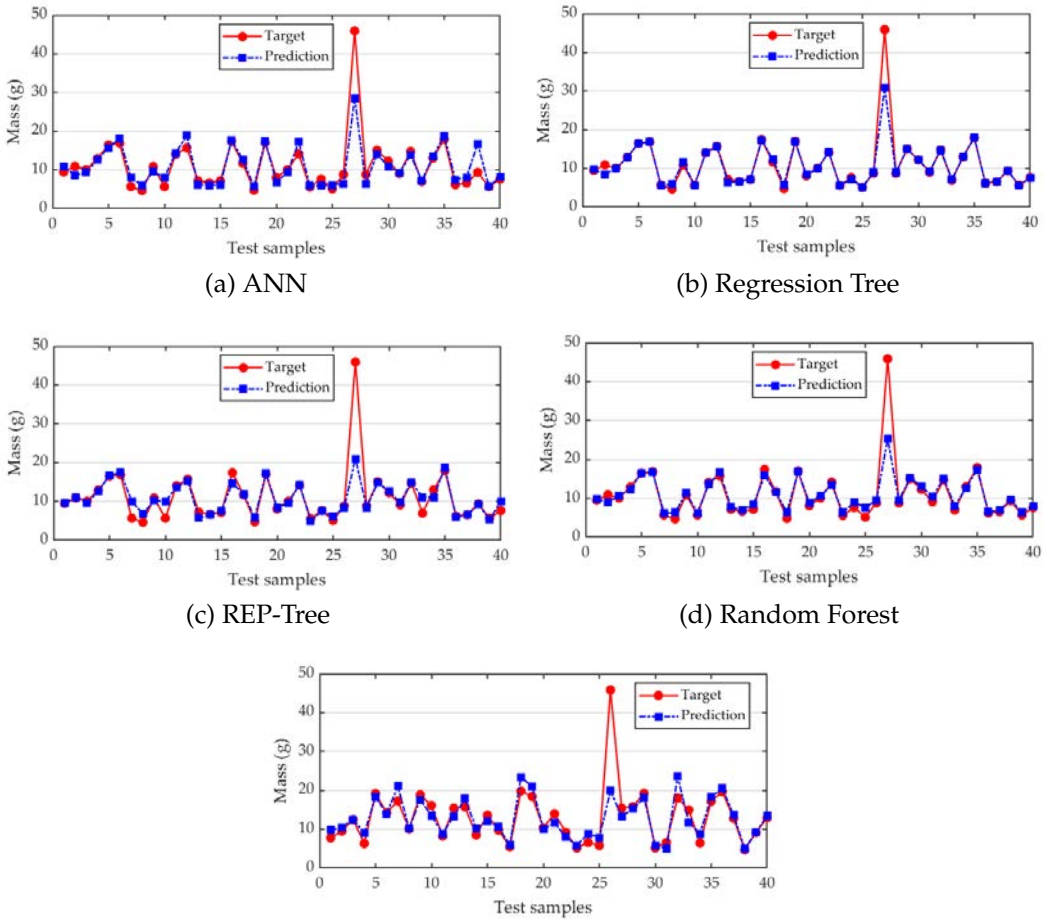
4.1 Model prediction analysis

After a data preparation step, the linear filling data-set is taken into account and fed into the Machine Learning tools to start a predictive cycle. The data-set is partitioned 10 times with a 10-folds cross validation method. At every split, the training set ($\mathcal{L}$) has 90 % of the initial and the testing set ($\mathcal{T}$) contains the remaining 10 % of the experiments. At every loop, the training results are saved and summed with the previous ones. Thus, the final testing set ha exactly the same amount of data of the initial data-set. For every fold, new performance indices are computed and eventually an average between 10 folds is displayed to the user. This logic procedure is clarified in Figure 3.1.
In addition to that, mass is stratified as a feature, hence every single initial data is used also as a goal in the testing set after the 10-folds cross validation method. Thanks to the *No Free Lunch Theorem* proposed by Wolpert and Macready in [34], we can state that it is impossible to know in advance which of the proposed predictive algorithm is going to learn better the data-set since every type of different technique is equally likely to score the same across the unlimited variety of possible learning problems. The only way to understand which one fits better the output is just using all the available techniques and compare their results.
In Table 4.1 a summery of all the proposed performance indices is shown for the linear die filling data-set. From these scores, it is evident that the Regression Tree

model is the Machine Learning technique that gives back the lowest error and the higher *correlation coefficient* and *determination coefficient*. Moreover, from the same results, it is also clear that ANN model is the worst models in terms of prediction performances, while REP-Tree and FNT models performs quite similarly. This is even more evidenced by Figure 4.1 and Figure 4.2 where the predicted output and the real output are compared with two different plot styles.
Here, it is visible that the Regression Tree model is very accurate in predicting the target and the predicted data basically overlap the real data in Figure 4.1 while all points almost lay in the bisector in Figure 4.2 and the slope of the regression line is the closer to one. Actually, with this learning technique the performance scores would have been even higher without the presence of some outliers around 40-50 g span. The presence of these points affects the final accuracy scores and also introduces some calibration errors in the model that produce a wrong prediction of other real data. These outliers are also depicted in Figure 4.1 by the bigger target peaks; as a matter of fact, these points are never learnt correctly by all employed algorithms.
From Figure 4.2 it is clear that also the data cluster is less dispersed and coarse for the Regression Tree while it is very broad for ANN and FNT models. REP-Tree and Random Forest shows an intermediate behaviour between Regression Tree and ANN models, however they still have an acceptable learning accuracy. Apparently, splitting the initial data-set into many sub-sets and then finding a regression rule for every sub-set is the best method to learn this specific problem concerning linear die filling. For the seek of completeness, the Regression Tree inner levels are set to a maximum value of 10.

Table 4.1: Performance indexes of Machine Learning predictive analysis carried out on the testing data for linear die filling. All indexes are an average between the values of 10 different folds provided by the 10-folds cross validation method.

Machine Learning Model	$\overline{\text{RMSE}}$	$\overline{R^2}$	$\overline{r}$	σ_r
Artificial Neural Network	2.5141	0.7514	0.8578	0.1312
Regression Tree	1.8513	0.8642	0.9209	0.1330
Random Forest	2.2757	0.8295	0.9026	0.1275
REP-Tree	2.4105	0.8083	0.8884	0.1447
FNT	2.5123	0.7834	0.8836	0.0544



(a) ANN

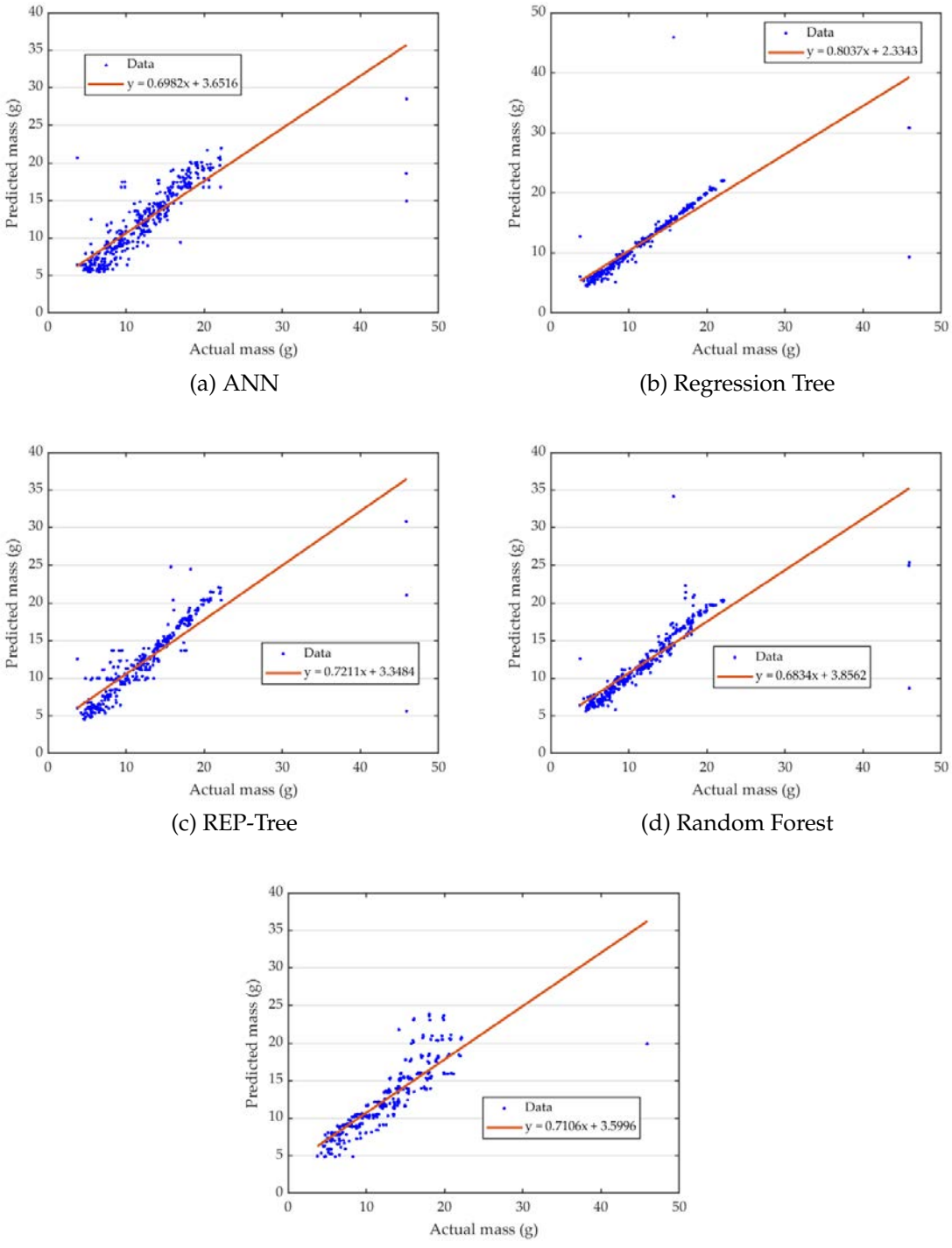
(b) Regression Tree

(c) REP-Tree

(d) Random Forest

(e) FNT

Figure 4.1: Target versus prediction plots of the real mass value for 40 samples randomly taken from the linear die filling data-set.



(a) ANN

(b) Regression Tree

(c) REP-Tree

(d) Random Forest

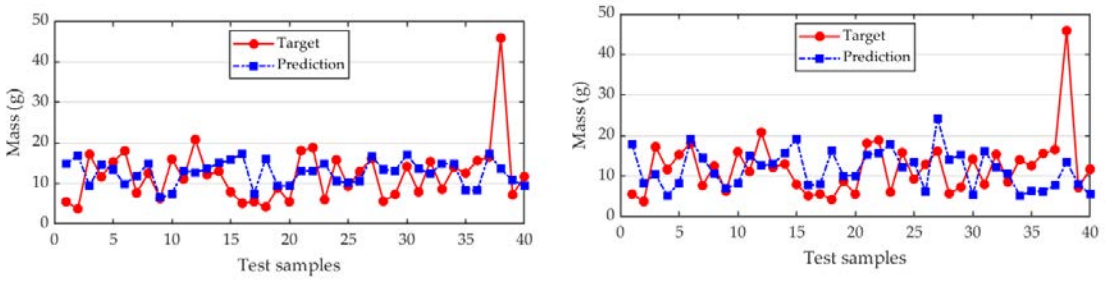
(e) FNT

Figure 4.2: Scatter plot between real mass and predicted mass for linear die filling.

4.2 Feature Analysis

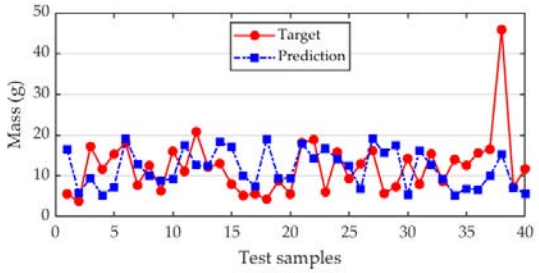
After the pure application of the already existing Machine Learning techniques, Regression Tree is taken as a reference model for the feature analysis. In fact, Regression Tree has proven to achieve the best predictive results, thus is taken as the best model to carry out a feature analysis on the four studied inputs: the true density (ρ_{true}), the median diameter of the size distribution (d_{50}), the upper diameter size of the considered material class (d_{max}) and the shoe speed (v_s).
We would like to implement a feature analysis in order to find out the most important and leading features that has the strongest impact in determining the mass output. To do so, we try to discard consequently all inputs and we save the input-reduced sub-set with the lowest RMSE and the higher r. Hence, the discarded input feature is always the last decisive for the outcome prediction. Generally speaking, a input-reduced data-set is harder to learn by a ML technique and the predictive analysis is going to be less and less precise. Again, a 10-folds cross validation method is supplemented in order to validate the final testing results.
From Table 4.2 it is clear that the first non important input feature is the true density of the material, this feature can easily be discarded without losing to much prediction accuracy. In fact, the true density gives has only 3 values for the entire data-set and gives only information on the type of material we are looking at, in this case it behaves more similarly to a categorical feature than to a numerical variable.
Furthermore, the median granule size is the following last important input. Also in this case, there are only 3 values for d_{50} in the whole data-set (one for each material) and this input seems to give information only about the material type rather than to other properties. In fact, the accuracy scores with 3 inputs are very similar to the one with only 2 inputs; thus the presence of d_{50} does not seem to provide many additional details to the learning process with respect to the true density. Besides, the two most important input feature are represented by the granule class upper size and by the shoe speed. Specifically, these two specifications affects the most the final mass deposited in the die. Between these two possibilities, the shoe speed (v_s) is the most important and significant input variable in the data-set in order to define and rule the final outcome. Indeed, we expect as a rule of thumb for every material that as the shoe speed increases, the

deposited mass decreases. From Fig. 4.3 and Fig. 4.4 it is evident that the prediction accuracy becomes less and less accurate as the data-set is further reduced. Despite being the best model for learning the linear die filling data-set, Regression Tree shows a lot of difficulties to predict the reduced sub-sets and the data cluster becomes more and more dispersed and the trend line slope becomes closer to 0 (Fig. 4.4). This is due to the loss of valuable information discarding the previous input feature and the hardship of the ML technique to predict an outcome with a data scarcity.

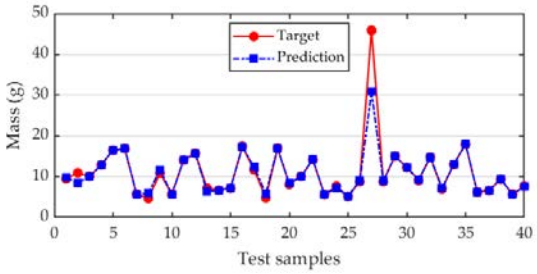


(a) 1 Feature

(b) 2 Features



(c) 3 Features

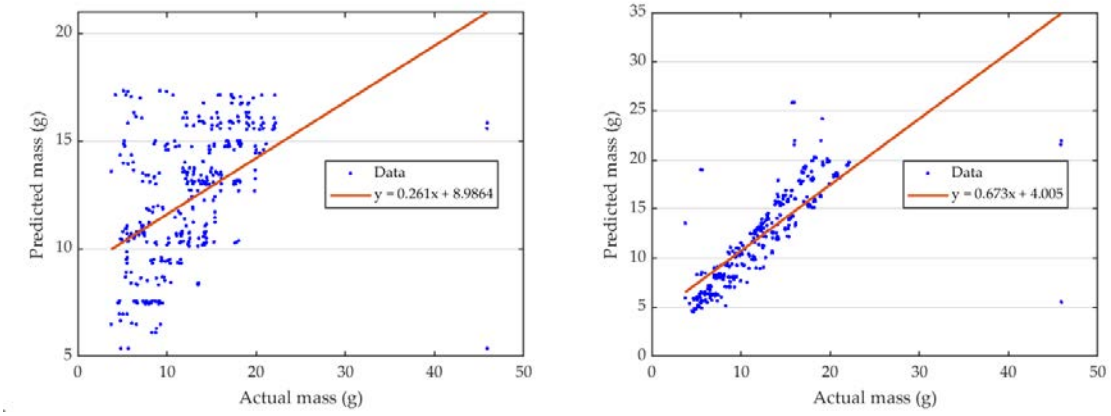


(d) 4 Features

Figure 4.3: Feature analysis performed by Regression Tree model. Target versus prediction plots of the real mass value for 40 samples randomly taken from the linear die filling data-set

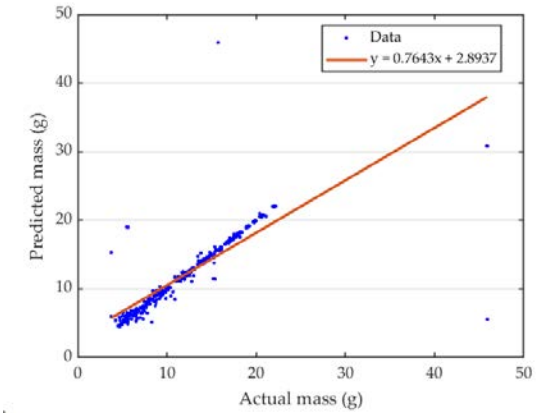
Table 4.2: Score indexes for feature analysis performed by Regression Tree model on the testing data. All indexes are an average between the values of 10 different folds provided by the 10-folds cross validation method.

N° of features	Selected features	$\overline{\text{RMSE}}$	$\overline{R^2}$	$\overline{r}$	σ_r
4	v_s, d_{max}, d_{50}, ρ_{true}	1.8513	0.8642	0.9209	0.133
3	v_s, d_{max}, d_{50}	2.4744	0.7929	0.8790	0.1495
2	v_s, d_{max}	2.9484	0.7050	0.8273	0.1501
1	v_s	4.6375	0.2889	0.5167	0.155

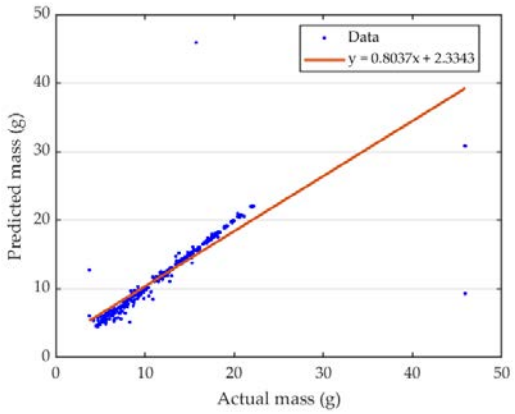


(a) 1 Feature

(b) 2 Features



(c) 3 Features



(d) 4 Features

Figure 4.4: Scatter plots for feature analysis performed by Regression Tree model.

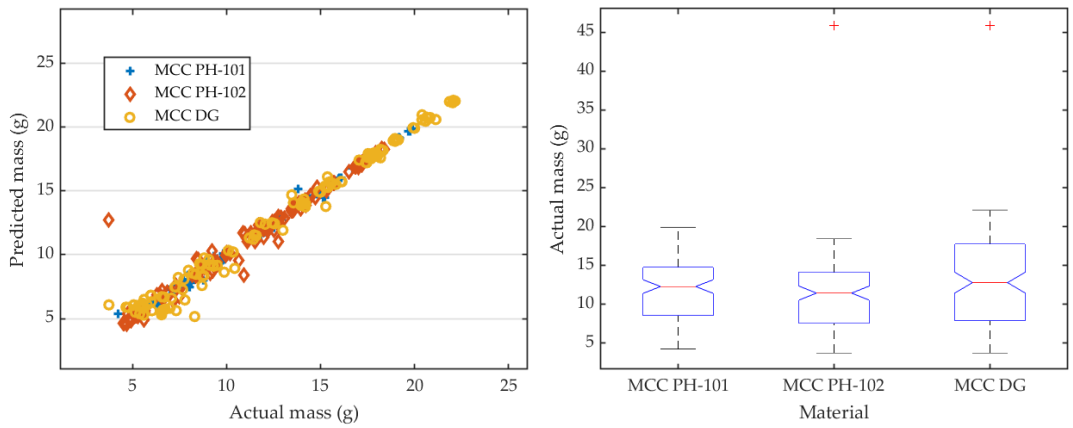
4.3 Data-set interpretation

Regression Tree model has proved to be sufficiently accurate to carry out a predictive analysis on the linear die filling data-set; it could have been even better without the presence of some outliers since the main pat of the data cluster is basically located on the 2D plane bisector of the scatter plots between the predicted data and the real data. Despite that, some further statistical information can be retrieved from the initial data to stress out some physical behaviour of the operation conditions with respect to the deposited mass in the die. In order to do so, some box plots are plotted from the initial data-set according to the real input feature they refer to. In addition to that, the points of the scatter plot referred to the Regression Tree performance analysis (Fig. 4.2) is divided and coloured by input features to visualise better the hidden impact of the input features on the final mass outcome.

To begin with, in Figure 4.5 data are divided according only to their material, this is simply achieved by splitting the data by their true density (ρ_{true}) values (or median size). As is evident, MCC DG data have an higher span in terms of final mass and its median value is higher than the other two materials. Hence, MCC DG tends to show a bigger mass deposit in the die with respect to MCC PH-101 and MCC PH-102 considering all upper granule sizes and shoe speeds. This is probably due to the fact that MCC DG has the best flowing properties through a hole as depicted in Table 3.2. MCC PH-102 may have the highest flow factor but MCC DG flows better when it comes to fall from a aperture. Therefore, we expect that materials with a high flow index and flow factor to fill more the die in a linear filling process. MCC PH-101 and MCC PH-102 shows similar results in terms of deposited mass even if they have quite different flow properties. In fact, their mass flow rate through an orifice is very similar.

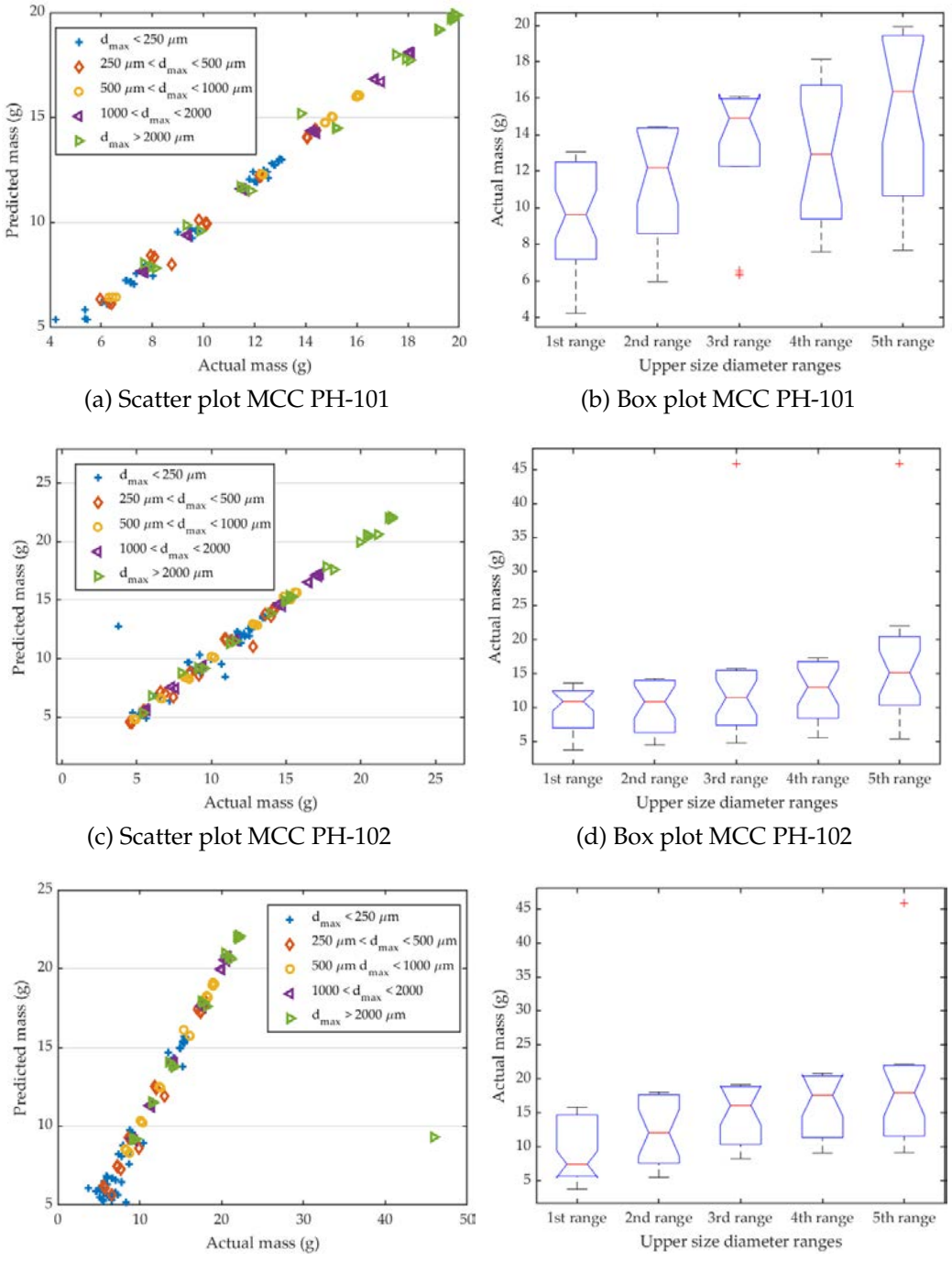
Moreover, data are divided and compared by their upper granule size limit (Fig. 4.6). Five ranges are binned for d_{max} looking at its distribution provided in the Appendix (Fig. A.2). It is very clear that the mass deposited increases as the granules becomes bigger. This is true for every analysed material and for all the considered shoe speeds. In facts, the bigger the particle, the weaker the strength of the capillary and electrostatic forces between granules, thus the less cohesive the powder. In a nutshell, the bigger the granule, the more free flowing the material, thus the die is more filled.

Eventually, in Figure 4.7 data are divided and coloured according to their real shoe speed values and material. In this case, four shoe speed ranges are identified looking at the histogram in Figure A.1. From Fig. 4.7 it is noticeable that the deposited mass decreases as the shoe speed increases for the three analysed materials. The experimental set-up quite always goes beyond the material critical velocities (v_c), so we expect that the die is almost never fully filled. Interestingly, the first shoe speed range shows a lower median deposited mass than the following range. However, this range has a very large span between the minimum and maximum value, so it is possible that this is a result of some data scarcity and, with more initial data, this median position may have been greater than the second median value. Another very interesting fact is that as shoe speed increases, the box plots tends to reduce their span; hence the final outcome tends to be more stable around a precise value as the shoe speed becomes faster.
It may be also important to underline that the model predicts more precisely the output feature for coarser granules than to finer granules for all material as reported in [20].
To sum up, MCC DG is the most free falling material that returns greater deposit in the die, bigger granules also more easily fall into the die since they are typically less cohesive, and higher shoe speeds give back lower filling ratios for every material since it has less time to pass through the die from the feed frame in the filling process.



(a) Scatter plot (b) Box plot

Figure 4.5: Scatter plot of linear die filling performed by Regression Tree and coloured by material. Box plot of the real output mass divided by material.



(a) Scatter plot MCC PH-101

(b) Box plot MCC PH-101

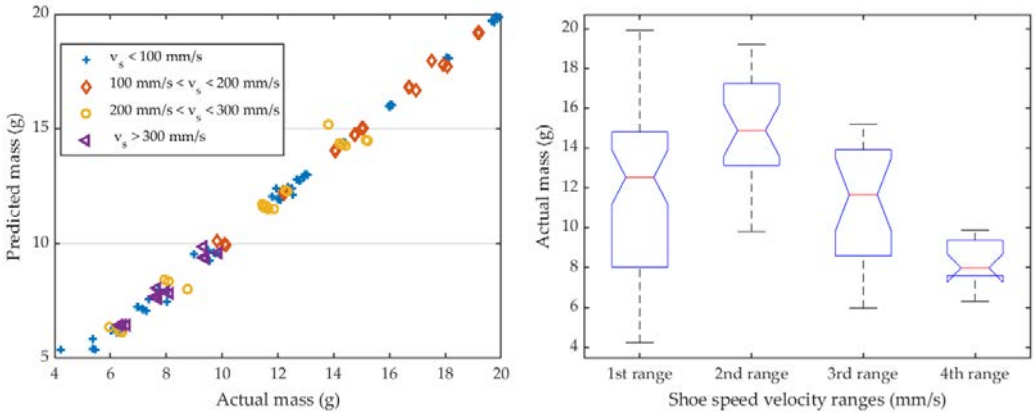
(c) Scatter plot MCC PH-102

(d) Box plot MCC PH-102

(e) Scatter plot MCC DG

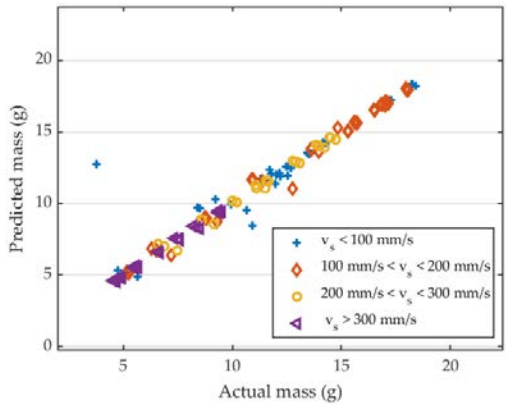
(f) Box plot MCC DG

Figure 4.6: Scatter plot of linear die filling performed by Regression Tree, coloured by upper size limit range (d_{max}) and divided by material. Box plot of the real output mass divided by upper size limit range and by material. Notches in box plots stand for a statistical uncertainty to identify the limit of the percentile.

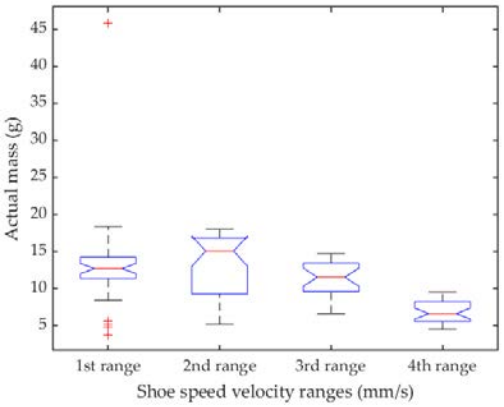


(a) Scatter plot MCC PH-101

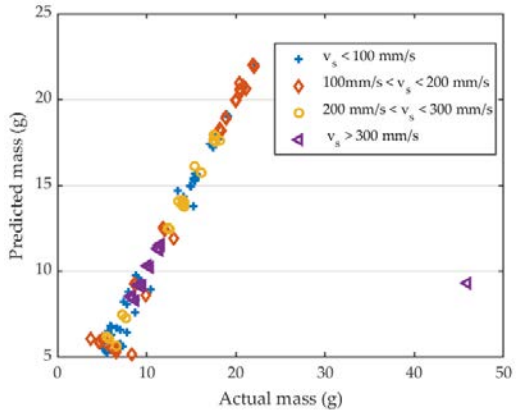
(b) Box plot MCC PH-101



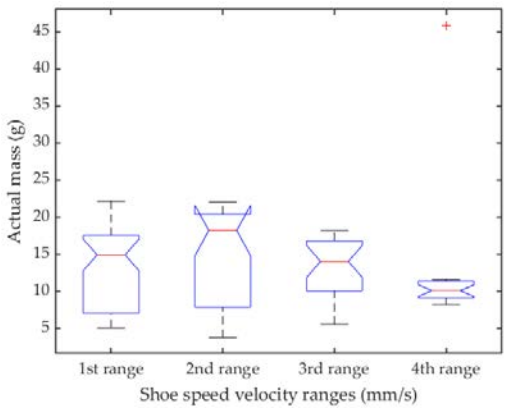
(c) Scatter plot MCC PH-102



(d) Box plot MCC PH-102



(e) Scatter plot MCC DG



(f) Box plot MCC DG

Figure 4.7: Scatter plot of linear die filling performed by Regression Tree coloured by shoe speed ranges (v_s) and divided by material. Box plot of the real output mass divided by upper shoe speed ranges and by material. Notches in box plots stand for a statistical uncertainty to identify the limit of the percentile.

Chapter 5

Rotary die filling: results and discussion

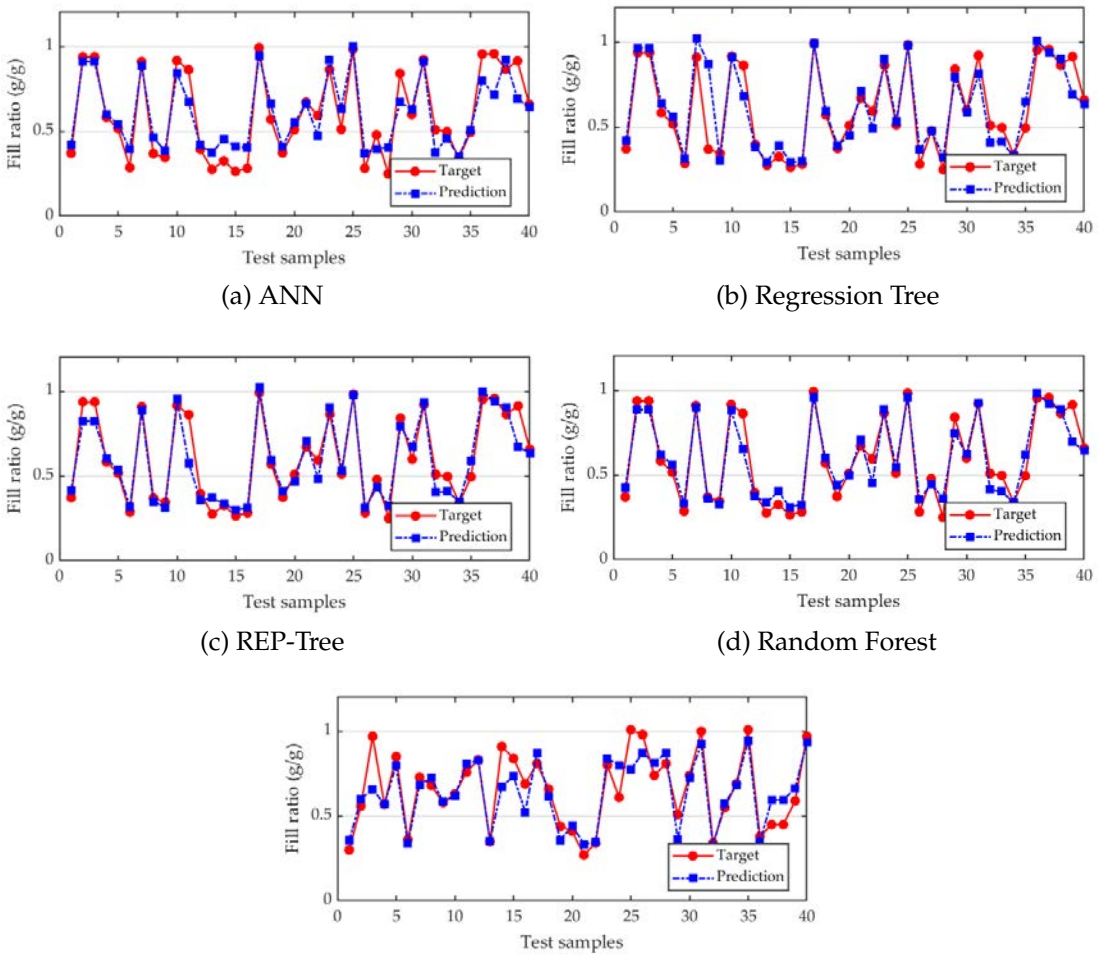
5.1 Model prediction analysis

The same calculation procedure described is Chapter 4 is applied for the rotary die filling data-set. After a preliminary data manipulation step, the data-set is partitioned into a training and testing data-sets with a 9:1 proportion, then the Machine Learning techniques are applied to the training set and the performance indices are computed for the testing data-set which defines the output target. Also in this case, a 10-folds Cross Validation method is used to justify the predictive analysis. Again, the data-set is partitioned stratifying the final output (fill ratio δ); therefore the final aggregated testing data-set corresponds exactly with the initial rotary die-filling data-set but split in 10 different folds.
Anew, *No Free Lunch Theorem* [34] suggests that it is impossible to know in advance which algorithm predicts better the target feature, hence all computational techniques need to be implemented to the data-set and their prediction results must be compared.
In Table 5.1 accuracy indices are displayed for the prediction analysis carried out on the rotary die filling issue. From this table, Random Forest appears to be the most suitable ML technique to learn this data-set since it shows the lowest RMSE and the highest *correlation factor* and *determination factor*. On the contrary, from these data the Flexible Neural Tree model struggles the most in order to

predict the outcome. It is also interesting to notice that in this case Random Forest has the lowest standard deviation related to r between the 10 folds while FNT presents the highest one. This suggests that Random Forests has the lowest variability in terms of learning results than all the other models; in other words, Random Forest proves to be a consistent algorithm for this problems across all 10 folds. On the other hand, FNT is a very variable and unstable model and it can learn adequately or poorly different testing sets.
After this technique, REP-Tree has the second best learning score among the used models, whereas Artificial Neural Network and Regression Tree display very similar prediction scores. From Figure 5.1 we can see that Random Forest is the approach that best overlaps the target fill ratio and FNT is the model that fails the most to foretell the output feature. It is also important to underline that Regression Tree seems to overall predict quite well the target but some points are awfully foretold, thus the global accuracy is not the best one. This is more evident from Figure 5.2, where the data cluster is the broader for FNT and the nearer to the trend line for Random Forest. We can also see that for Regression Tree data are not that broad but many points are located far away from the main body of the cluster and this affects a lot the final prediction accuracy. From the same Figure, REP-Tree trend line slope is the closer to 1; however, the data cluster is quite broad so it is not the best model to use for learning this data-set.
From these evidences, a further implementation of a Regression Tree is needed in order to find the best predictive algorithm. It can be either a tree-branch pruning development such as REP-Tree or an ensemble method like the Random Forest model, in fact this data-set requires a more difficult approach to regress it properly.
Differently from the previous data-set, this is probably due to the fact that here input features affects the outcome more equally than the first data-set and, in order to find the best input sub-set splits, a simple Regression Tree is not enough.

Table 5.1: Performance indexes of Machine Learning predictive analysis carried out on the testing data for rotary die filling. All indexes are an average between the values of 10 different folds provided by the 10-folds cross validation method.

Machine Learning Model	$\overline{\text{RMSE}}$	$\overline{R^2}$	$\overline{r}$	σ_r
Artificial Neural Network	0.1013	0.8145	0.9022	0.0245
Regression Tree	0.1005	0.8169	0.9013	0.0699
Random Forest	0.0648	0.9254	0.9619	0.0088
REP-Tree	0.0770	0.8926	0.9446	0.0162
FNT	0.1045	0.7887	0.8878	0.0261



(a) ANN

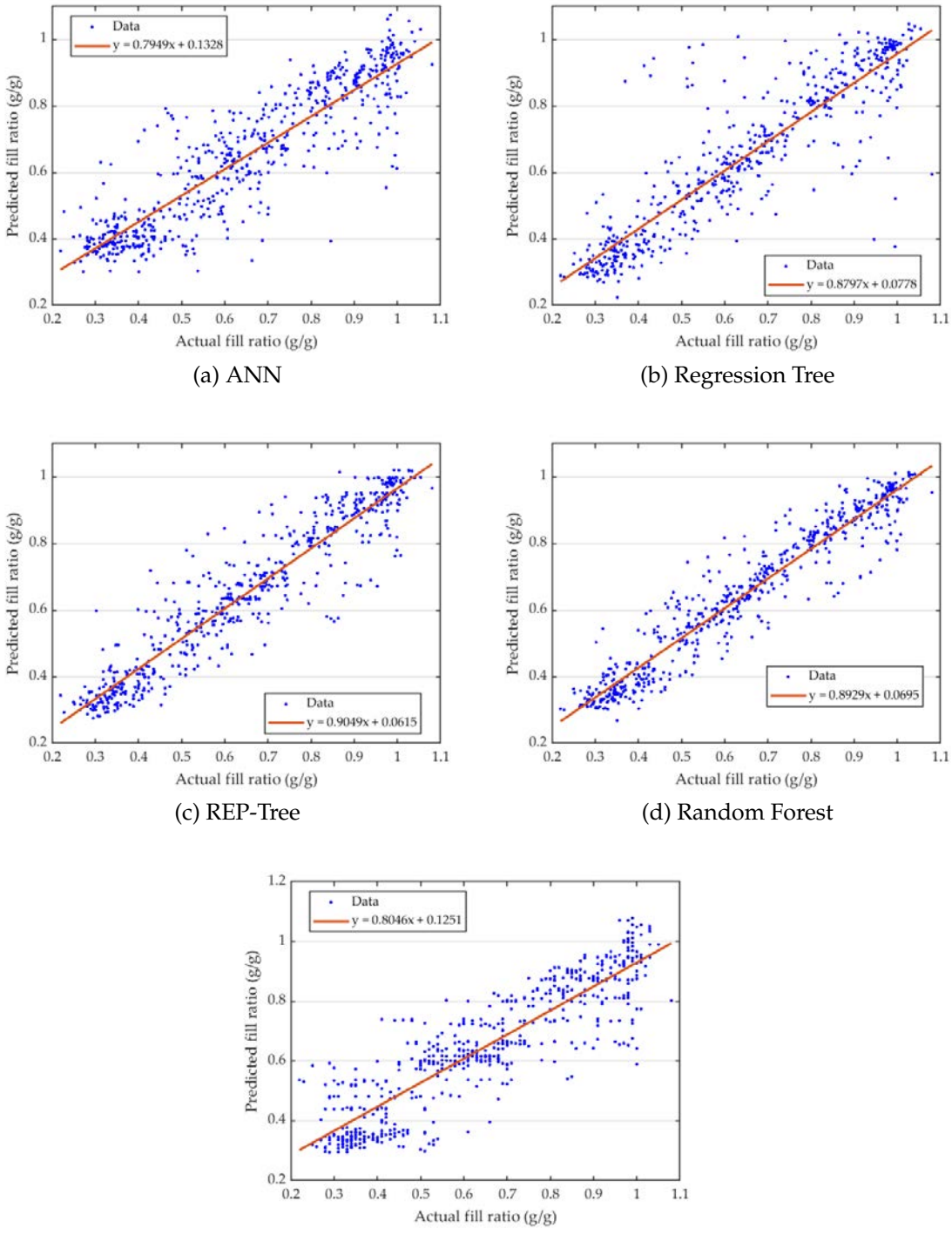
(b) Regression Tree

(c) REP-Tree

(d) Random Forest

(e) FNT

Figure 5.1: Target versus prediction plots of the real fill ratio value for 40 samples randomly taken from the rotary die filling data-set.



(a) ANN

(b) Regression Tree

(c) REP-Tree

(d) Random Forest

(e) FNT

Figure 5.2: Scatter plot between real fill ratio and predicted fill ratio for rotary die filling.

5.2 Feature analysis

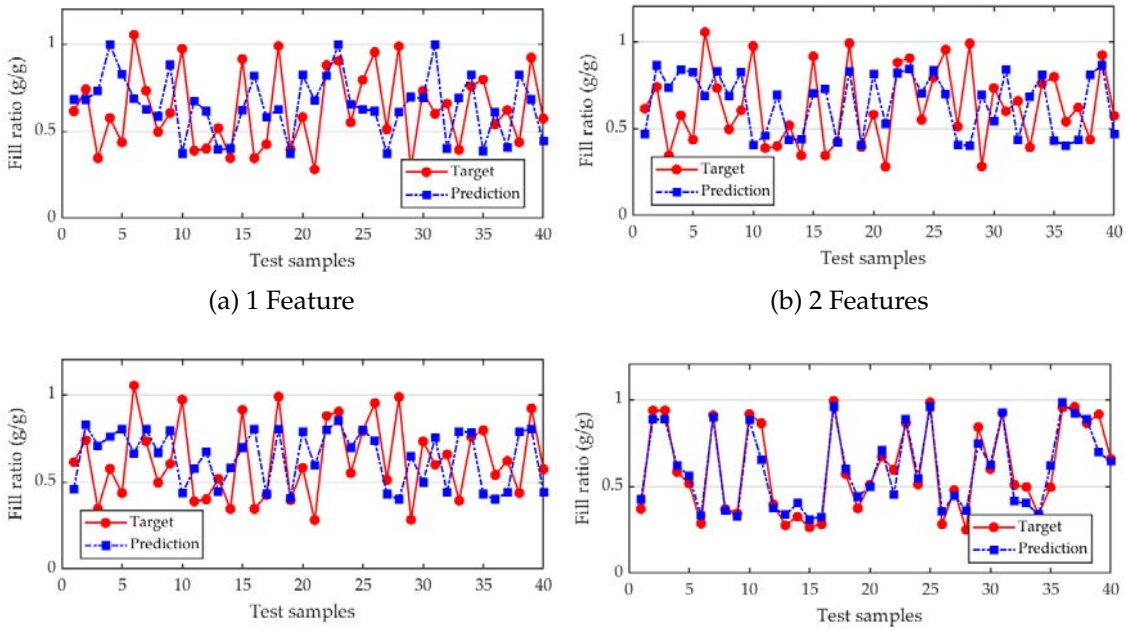
After the application of the five Machine Learning techniques described in the previous section, Random Forest is singled out as the reference models to carry out the feature analysis. In fact, it is not still clear from the scientific literature which is the most significant and important input variable in terms of die filling performances during a rotary die filling. In addition to that, it is not even clear if critical velocities, derived from linear die filling experiments, can be used as a term of comparison in rotary die filling to understand the pivotal turret velocity over which dies are never filled up. Since Random Forest shows in Table 5.1 the best predictive results, it is also used for learning the input-reduced data-set since it is still expected to return the best predictive analysis.
Again, the data-set is cyclically input-reduced and the sub-set with the best accuracy is confirmed while inputs with a lower guessing weight are gradually discarded from the data. Therefore, a sort of importance ranking is built up amongst all input variables and it is expected that only the most significant and important input feature remains to be learnt by the ML model. Gradually, the reduced data-set loses many information and the algorithm struggles a lot to find hidden relationships for outcome prediction.
All feature analysis accuracy indices are shown in Table 5.2 whereas in Figure 5.3 and Figure 5.4 we find the prediction versus target plots and the scatter plots related to the reduced data-set. From these results, it is evident that just getting rid of only one input variable implies increasing a lot the prediction error and worsening significantly the overall accuracy. This suggests that in this specific case every feature weights more in terms of outcome impact than the linear die filling case.
To begin with, paddle speed (v_p) is the first withdrawn input variable. Despite that, this removal generates a strong downturn for the model prediction capability; Random Forest gives back inaccurate and broad data cluster as shown in Figure 5.4c. In addition to that, the trend line slope decreases unquestionably.
Furthermore, the n° die is pointed out to be the following last important variable of the data-set. Nevertheless, in this case the accuracy deterioration is amplified but not that much as the first removal. This may indicate that the two remaining input features, namely the Material ID and the turret speed (v_t), are more significant and incisive than the n° die for the output definition.

Nevertheless, in Fig. 5.4b the scatter plot of the reduced data-set related to only the Material ID and v_t shows that the prediction is utterly unreliable. Random Forest is not capable to foretell the fill ratio with a reasonable error span. In this case, the correlation factor is quite high ($\bar{r}$ = 0.8767) but it is evident that the model predicts data only within certain boundaries and not in the whole domain. Hence, the scatter plot appears to be a sum of following stripes and, instead of having a unique data cluster, many cluster appears in the scatter plot. This is an important warning for model unreliability. With only two features, the Random Forest becomes completely inadequate to picture the relationship between the reduced data-set and the fill ratio results. Nevertheless, the situation becomes even worse with only one input feature. The sole input variable that survives the feature elimination is the Material ID, which stands out as the most significant input. In this specific problem, the Material ID is treated more likely as a categorical value rather than a numerical value since it points out the material class the data refers to.

Apparently, the fill ratio (δ) final value is most affected by firstly the material type and, secondly, by the turret speed (v_t). However, differently from the previous data-set, in this rotary die filling question each variable seems to play a key role in determining the model predictive success. For instance, we expect that in linear die filling data-set the presence of d_{50} is quite negligible for the final accuracy score since the true density (ρ_{true}) suggests all the information d_{50} carries on. On the contrary, in rotary die filling data-set it is very hard to get rid of a single input feature without losing to much accuracy goodness.

Table 5.2: Score indexes for feature analysis performed by Random Forest model on the testing data. All indexes are an average between the values of 10 different folds provided by the 10-folds cross validation method.

N° of features	Selected features	$\overline{\text{RMSE}}$	$\overline{R^2}$	$\overline{r}$	σ_r
4	Material ID, v_t, n° die, v_p	0.0648	0.9254	0.9619	0.0088
3	Material ID, v_t, n° die	0.1122	0.8332	0.9127	0.0144
2	Material ID, v_t	0.1162	0.7689	0.8767	0.0157
1	Material ID	0.1461	0.6019	0.7757	0.0169



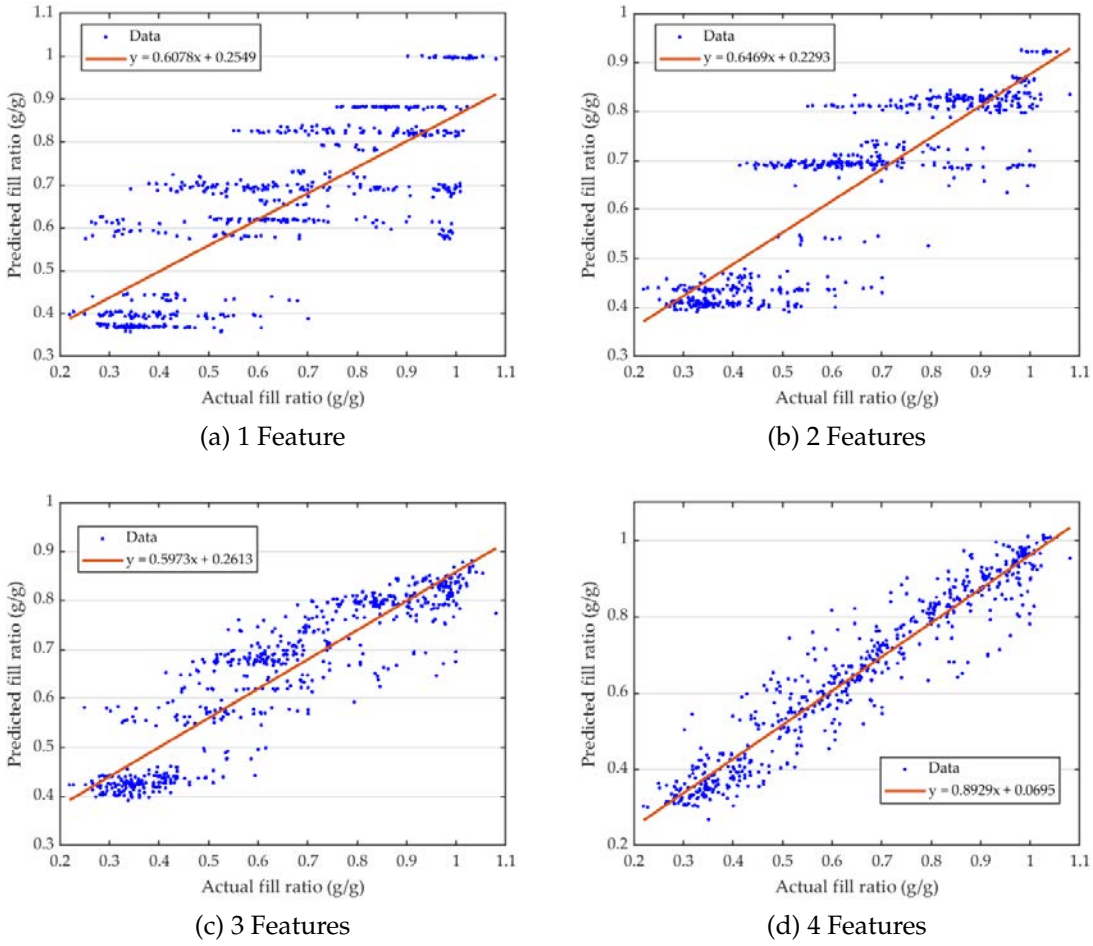
(a) 1 Feature

(b) 2 Features

(c) 3 Features

(d) 4 Features

Figure 5.3: Feature analysis performed by Random Forest model. Target versus prediction plots of the real fill ratio value for 40 samples randomly taken from the rotary die filling data-set.



(a) 1 Feature

(b) 2 Features

(c) 3 Features

(d) 4 Features

Figure 5.4: Scatter plots for feature analysis performed by Random Forest model.

5.3 Data-set interpretation

Random Forest model has proved its capability to learn satisfactorily the rotary die filling data-set and to predict the testing sub-set with a slightly coarse data cluster. With respect to linear die filling data-set, this predicted data group is broader; however, in this case there are not any type of outliers and the overall prediction scores for this technique are sufficiently high and acceptable. Maybe, a more refined, advanced and calibrated type of Regression Tree algorithm may show a better prediction since all input features interacts a lot more with each other in this specific problem. Due to that, the initial data needs to be split in

regressed very carefully in order to learn and find out the hidden variable relationships to predict the fill ratio outcome. Moreover, tree models usually are non-robust because a small change in the training set can cause a notable change in tree structure, thus also in the final prediction. Hence, in this case Random Forest is likely to be affected by the large variability in the 10 training folds of the cross validation method, resulting in predicting similar input features in slightly different results and creating a quite broad prediction cluster. Besides, here *overfitting* does not seem to occur since the variables interactions are not that trivial. In fact, REP-Tree does not provide a further improvement for the data prevision. A very advance future possibility to learn this data-set is to apply a *Rotation Forest*. Basically, it is an ensemble framework where each regression tree in the forest is initially trained by the application of *Principal Component Analysis* (PCA) on a random sub-set of the input features [35].

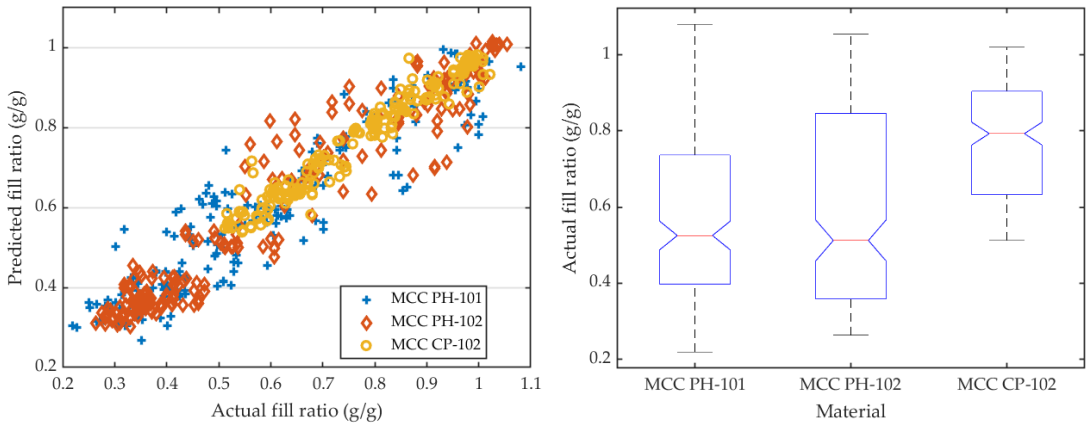
For the sake of clarity, it may be useful to perform a statistical analysis on the initial real data to understand if there are any kind of physical behaviours that can be observed. In order to do so, the scatter plots from Fig. 5.2 are coloured by real input reference and divided by material; in addition to that, also some box plots are carried out on the real data to see the influence of every input variable on the unique final outcome.
To start with, real data are split according only to their material in Figure 5.5. It is very clear from this evidence that MCC CP-102 is the material that returns the highest fill ratio span considering all turret and paddle speeds and all die positions. This is clearly due to the better flow properties of this material from an aperture than MCC PH-101 and MCC PH-102. In fact, from Table 3.4 MCC CP-102 has the lowest flow index and the larger mass flow rate through an orifice. Despite not having the larger flow function, this spheroidal material is the most free flowing from an open hole of a feed frame. This material has also the highest linear critical velocity (v_c=148.2), so dies are expected to be more filled with lower turret speeds than the other two powders. Moreover, we take note from the same table that MCC PH-101 should be the most cohesive powder but in reality it has a fill ratio span very similar to MCC PH-102 one and, like the previous data-set, there are not filling differences between these two materials in rotary die filling problems.
Besides, in Figure 5.6 all data are divided according to their reference material

and coloured by the n° die in the turret table. It is worth noting that for all the studied materials there are no clear filling disparities between the 1st, 2nd and 3rd die in the turret table, whereas the 4th die results slightly more filled than the previous three ones. As reported in [16], this is explained by the fact that at the beginning of the filling process there is a thin compact powder layer between the feed frame and and turret table due to the paddle rotation. When the turret spins and the dies appears under this thin and thick layer (5 mm), it inhibits at some extent the mass flow rate through the aperture. This resistance effect is overcome after the 3rd die, the thick layer dies out and the flow rate performances increase as well as the fill ratio. This effect is present in all the data disregarding the material class. Of course, MCC CP-102 filling behaviour has the same trend across the four dies but with higher fill ratio values.

Furthermore, the influence of the paddle speed is examined in Fig. 5.7. In this figure, real data are split according to the five used paddle speed (v_p). The effect of this input feature is quite complicated and hard to explain. Paddle imposes a shear stress on the material in the feed frame both while the discharge area is open and closed. From Figure 5.7, it is noticeable that for MCC PH-102 the effect of the paddle speed is totally negligible. This material is indeed classified as a free flowing material according to its flow function (ff) and the rotation speed of a paddle does not provide any further enhancement on the filling performances. The fill ratio may be improved by the addition of some lubricants in the feed frame, which can significantly improve the fill ratio with the paddle speed increase as reported in [36]. Despite that, it is shown that as the paddle speed increases, the fill ratio slightly improves for more cohesive material like MCC PH-101 and MCC CP-102. The most evident enhancement is reported for MCC PH-101, where the median fill ratio increases up to 0.65 for 50 rpm. It seems that over 50 rpm, a new steady state is found for both materials. Thus, it is expected that the paddle motion enhances the filling performances for more cohesive powders thanks to the applied shear stress in the feed frame but also that there is a critical paddle speed over which no more filling improvements are observed.

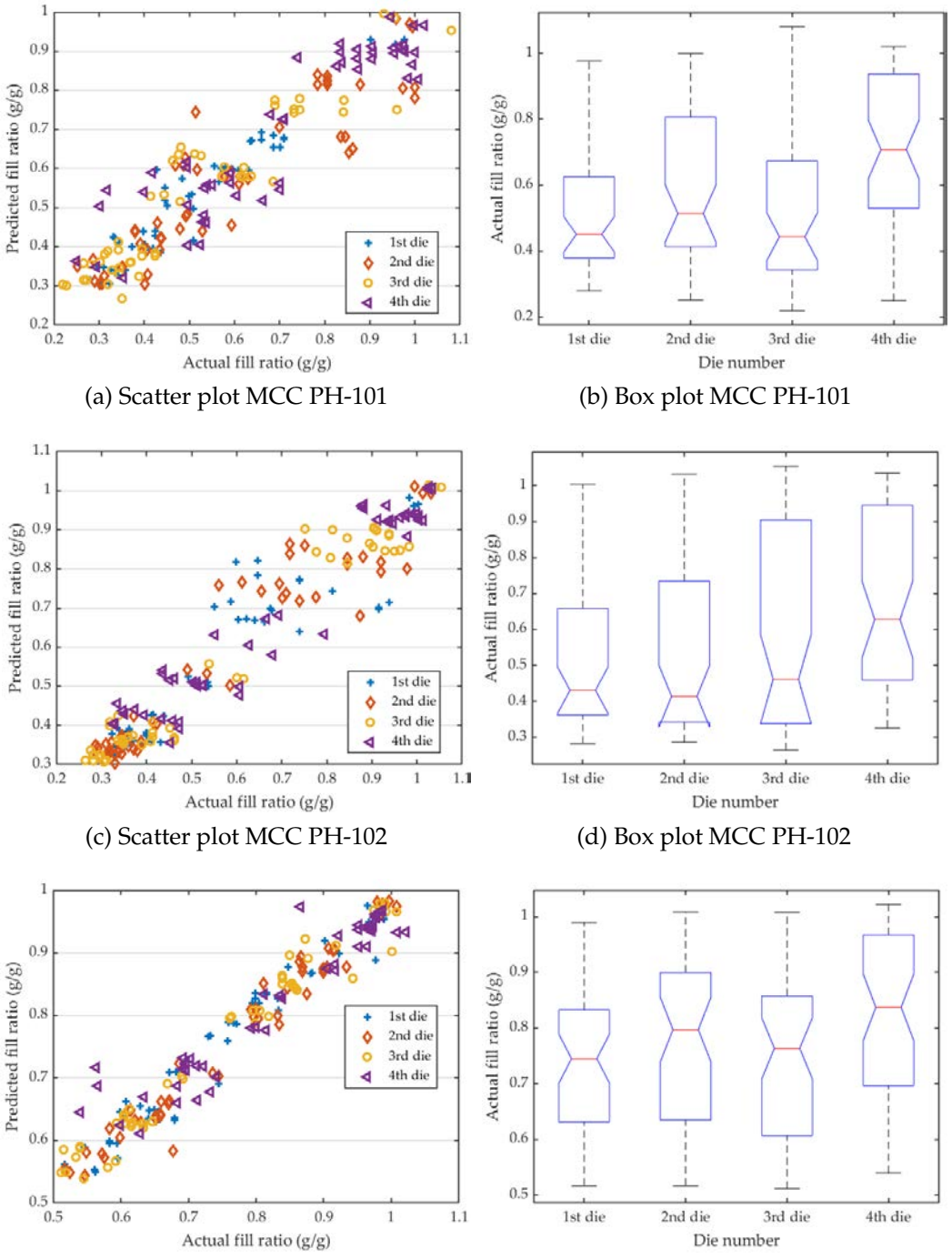
Eventually, the effect of the turret speed is reported in Figure 5.8. Here velocity ranges are binned looking at the turret speed distribution available in Figure A.3. This interpretation is very straightforward and it is consistent with the initial hypothesis on this behaviour. Namely, as the turret speed increases, the fill

ratio decreases exponentially. In fact, if the turret speed becomes higher, there is less time available for the material to flow from the feed frame to the die. Therefore, it is expected that the deposited mass reduces as it has less time to fill a void. This exponential behaviour is true and strongly evident for MCC PH-101 and MCC PH-102, whose linear critical velocities are $v_{c,PH-101}$=28 mm/s and $v_{c,PH-102}$=50 mm/s. These two velocities are easily passed over by the applied turret speed, so that the exponential decay is more visible. On the contrary, MCC CP-102 has a linear critical velocity equal to 148.2 mm/s, hence the exponential decay is less palpable since there are more applied velocities under this performance threshold. Moreover, MCC PH-102 has more rounded and free flowing granules from an aperture and this enhances the fill ratio performance even at high filling speeds. So that, the exponential decay is less pronounced.
At last, it really seems that the linear critical velocity can be used a reference term also for rotary die filling to have a hint for the critical velocity over which the die is always partially filled. To sum up, in rotary die filling MCC CP-102 is the most free flowing material that gives back a larger fill ratio with equal turret and paddle speeds. The 4th die always results more filled than the previous three due to the breakage of the material thin layer between the feed frame and the turret table. The paddle speed seems to slightly enhance only the filling performances of cohesive powders and, lastly, as the turret speed increases over the critical velocity, the die are exponentially less filled.



(a) Scatter plot (b) Box plot

Figure 5.5: Scatter plot of rotary die filling performed by Random Forest and coloured by material. Box plot of the real output fill ratio divided by material.



(a) Scatter plot MCC PH-101

(b) Box plot MCC PH-101

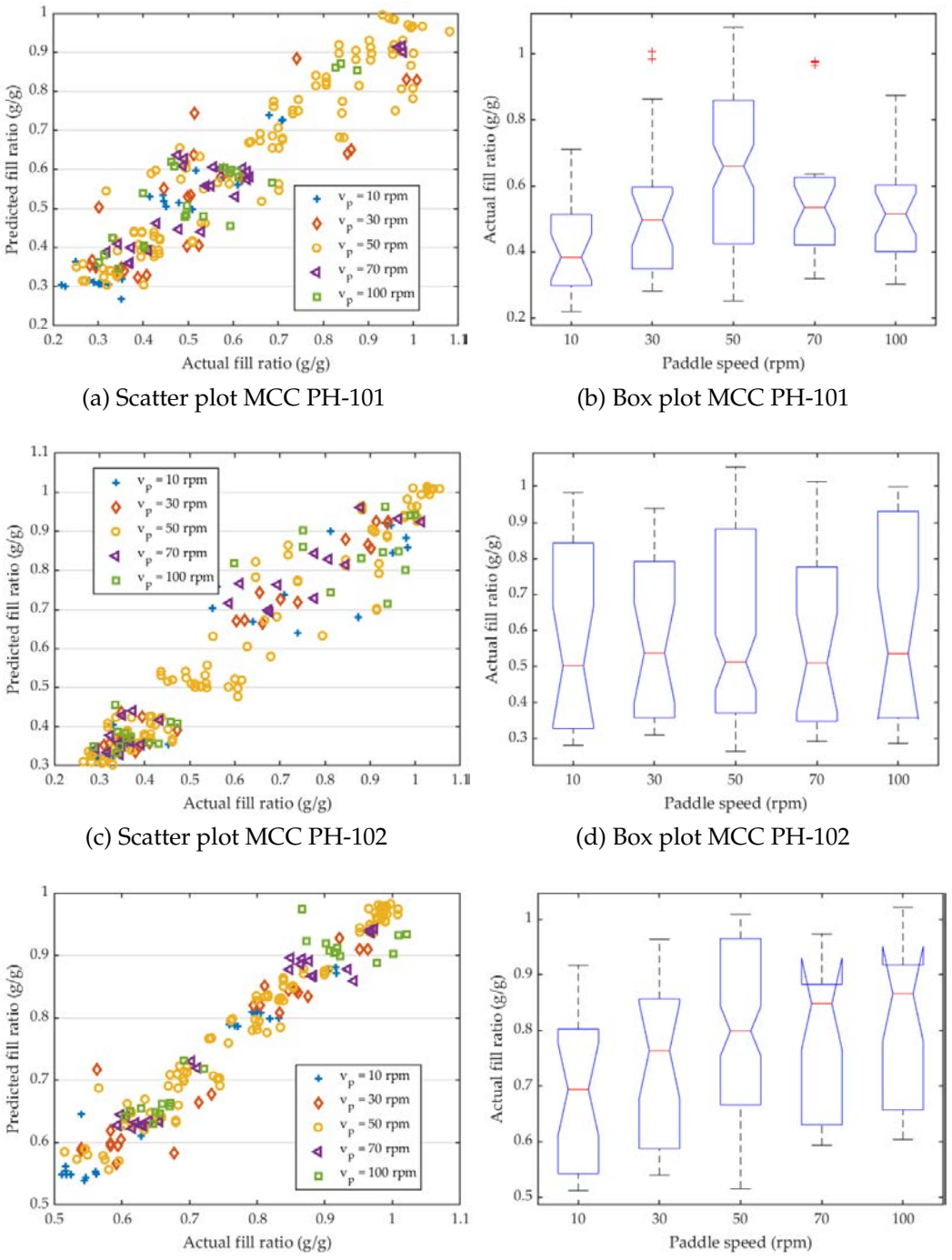
(c) Scatter plot MCC PH-102

(d) Box plot MCC PH-102

(e) Scatter plot MCC CP-102

(f) Box plot MCC CP-102

Figure 5.6: Scatter plot of rotary die filling performed by Random Forest, coloured by die number and divided by material. Box plot of the real output fill ratio divided by die number and by material. Notches in box plots stand for a statistical uncertainty to identify the limit of the percentile.



(a) Scatter plot MCC PH-101

(b) Box plot MCC PH-101

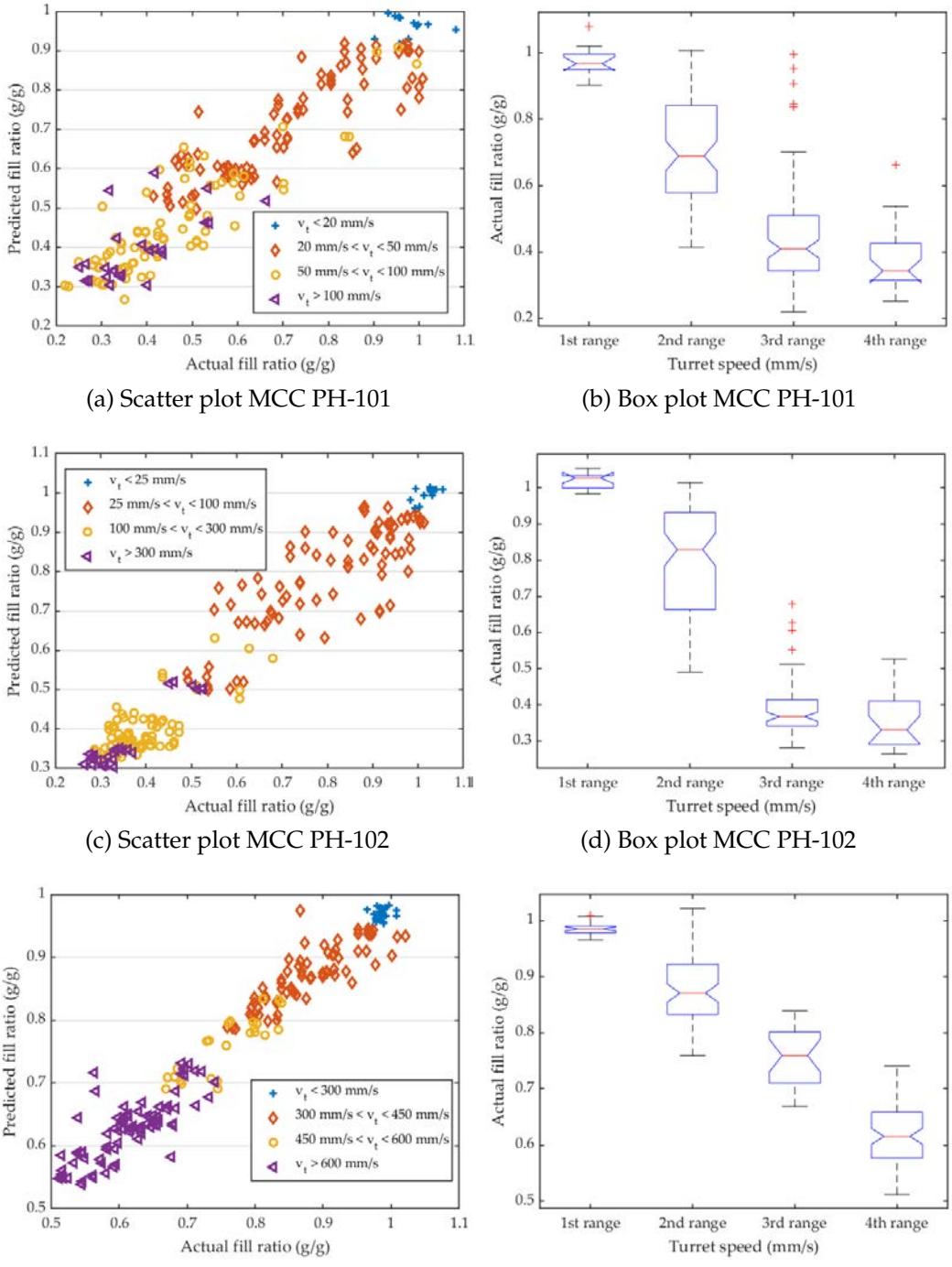
(c) Scatter plot MCC PH-102

(d) Box plot MCC PH-102

(e) Scatter plot MCC CP-102

(f) Box plot MCC CP-102

Figure 5.7: Scatter plot of rotary die filling performed by Random Forest, coloured by paddle speed (v_p) and divided by material. Box plot of the real output fill ratio divided by paddle speed and by material. Notches in box plots stand for a statistical uncertainty to identify the limit of the percentile.



(a) Scatter plot MCC PH-101

(b) Box plot MCC PH-101

(c) Scatter plot MCC PH-102

(d) Box plot MCC PH-102

(e) Scatter plot MCC CP-102

(f) Box plot MCC CP-102

Figure 5.8: Scatter plot of rotary die filling performed by Random Forest, coloured by turret speed (v_t) and divided by material. Box plot of the real output fill ratio divided by turret speed and by material. Notches in box plots stand for a statistical uncertainty to identify the limit of the percentile.

Chapter 6

Conclusions

This study considers two different experimental data-sets regarding linear die filling and rotary die filling for the tablet manufacturing operations. Four different grades of micro-crystalline cellulose are used to fill two types of die equipment, one linearly moving feed frame over a single die and one turret table equipped with four dies spinning under a rotating feed frame.
Since there are still not clear and consistent empirical and theoretical equations to predict the deposited mass into the die from the operation conditions, it is of interest to feed the data into an Artificial Intelligence algorithm to try to find out hidden features relationships.
Therefore, the supplied data-sets are learnt by five different Machine Learning techniques in order to find out the best predictive model. Eventually, some statistical analyses are carried out on the initial real data to understand some likely physical correlations between input and output variables.

Regarding linear die filling, this study points out Regression Tree as the best Machine Learning technique to predict the industrial application from the given input features. It is also found that the most significant and weighting input variables for determining the final deposited mass are the shoe speed (v_s) and the granule class upper size (d_{max}).
It is worth noticing that as the granules becomes bigger, the deposited mass increases and as the shoe speed increases, the die is less filled. Among the analysed materials, MCC DG has proved to be the most free flowing and shows the highest deposited mass span. In addition to that, one between the true density (ρ_{true}) and the median granule size (d_{50}) seems to be redundant in terms of information

to foretell the final outcome.
At last, it should be of interest to learn again the data-set discarding the possible marked outliers that are present inside the data. This way, the final prediction is expected to be even better and almost perfectly accurate with a Regression Tree model.

Moreover, the rotary die filling data-set does not seem so easy to learn by the analysed Machine Learning techniques. Here, all input features count and weight more equally on the final outcome determination and splitting properly the data-set with a Decision Tree enhanced model is not that easy. As a matter of fact, only Random Forest has proved to be the best predictive model across the five applied; however, it is also the most advanced and difficult to calibrate. Therefore, maybe one of the future perspectives is to ascertain that some further improved ensemble algorithm might anticipate more accurately the unique supplied output.
Feature analysis figures out that the Material ID and the turret speed (v_t) are the two most significant and important inputs for affecting the fill ratio. Amongst the studied powders, MCC CP-102 displays the best free flowing properties from an aperture and, as a consequence, it is the material that owns the highest fill ratio span with equal turret and paddle speed.
Besides, the statistical analysis shows that for every material the 4th die is always more filled than the three previous ones thanks to the breakage of the thin thick layer between the feed frame and turret table and that as the turret speed increases, the fill ratio decreases exponentially.
It's hard to tell how the paddle speed (v_p) affects the three material but it is shown its growth can facilitate cohesive powders to raise the fill ratio inside the die. On the contrary, its contribution is quite negligible in the feed frame for very free flowing powders.

Eventually, this study suggests that Machine Learning techniques can successfully learn die filling issues and foresee the unique numerical output related to the deposited mass. As a rule of thumb, for both problems large granules and free flowing powders fill the die with a larger amount of mass.
Therefore, this framework seems quite reassuring for industrial purposes in the pharmaceutical field. Machine Learning techniques can be successfully applied

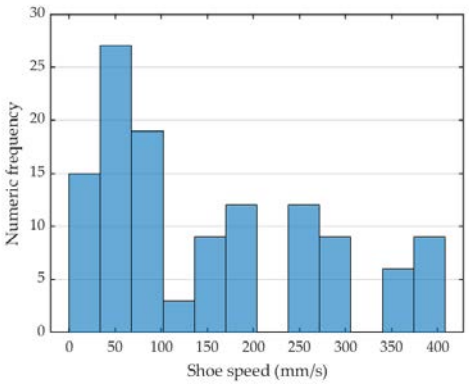
to real data to predict satisfactorily the filling target thanks to an external computational model.

Future work

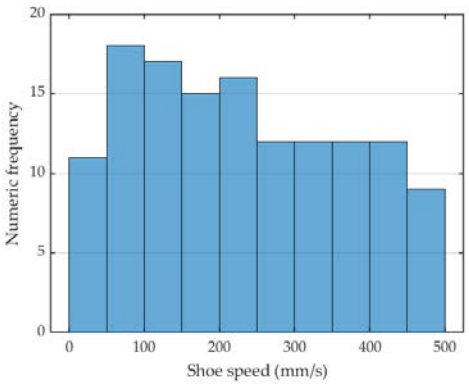
As a future perspective, a more robust and accurate algorithm must be found for rotary die filling problems in order to foretell the outcome in a more reliable confidence range.
In addition to that, the two filling problems should be worsened by adding more numerical outputs and see if the applied techniques are still reliable and accurate when they must learn and regress multiple outputs (e.g. fill ratio and blend composition).
Finally, two specific hand written Machine Learning algorithms should be pursued for the two problems starting from the Regression Tree and the Random Forest basic schemes. This way, two very problem-specific models are obtained to be well calibrated for the best possible problem predictability.

Appendix A

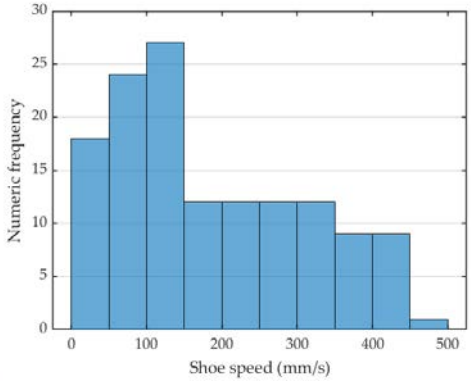
Appendix



(a) MCC PH-101

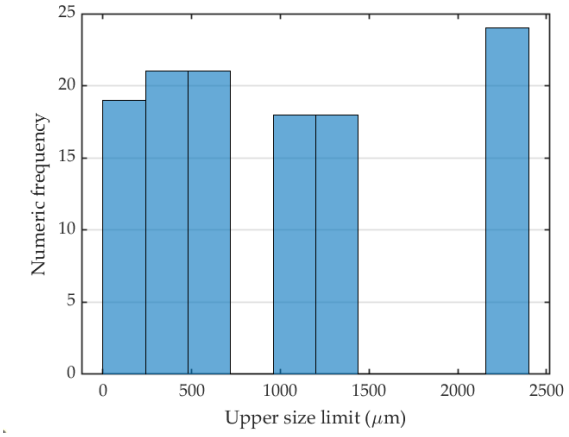


(b) MCC PH-102

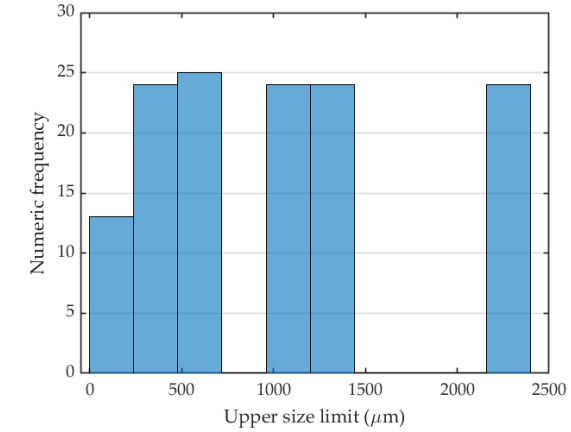


(c) MCC DG

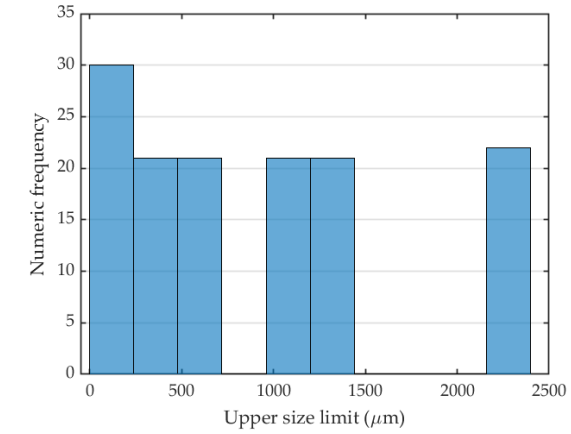
Figure A.1: Distribution of shoe speed for linear filling divided by materials.



(a) MCC PH-101

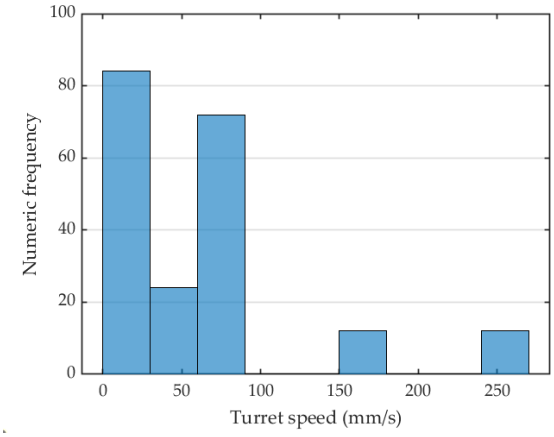


(b) MCC PH-102

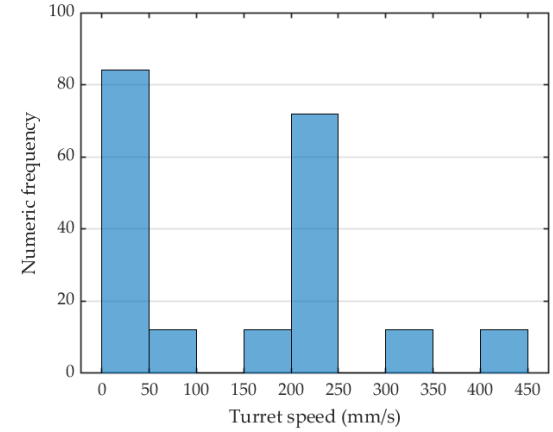


(c) MCC DG

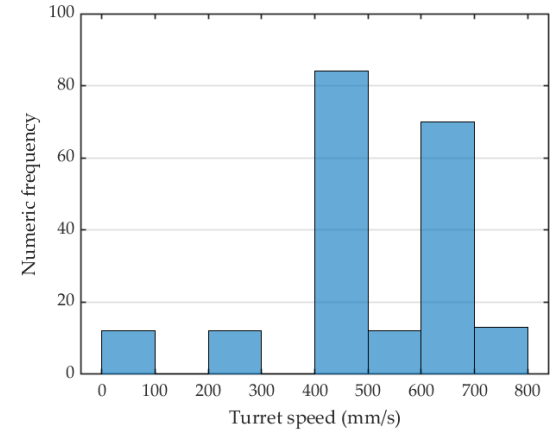
Figure A.2: Distribution of upper size limit for linear filling divided by materials.



(a) MCC PH-101



(b) MCC PH-102



(c) MCC CP-102

Figure A.3: Distribution of turret speed for rotary filling divided by materials.

Bibliography

[1] C. Y. Wu, L. Dihoru, and A. C. F. Cocks. "The flow of powder into simple and stepped dies". In: *Powder Technology* 134.1 (2003), pp. 24–39. ISSN: 0032-5910. DOI: 10.1016/S0032-5910(03)00130-X. URL: http://www.sciencedirect.com/science/article/pii/S003259100300130X.

[2] Martin Rhoades. *Introduction to Particle Technology*. 2nd edition. John Wiley & Sons, Ltd, 2008. DOI: 10.1002/9780470727102.ch1. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470727102.ch1.

[3] Andrea Claudio Santomaso. "Course slides - Characterization of Particulate solids". In: *Course: Particle Technology for the Food and Pharmaceutical Industries* University of Padova (2019).

[4] Andrea Claudio Santomaso. "Course slides - Distributed and Bulk Properties". In: *Course: Particle Technology for the Food and Pharmaceutical Industries* University of Padova (2019).

[5] Richard Holdich. *Fundamentals of Particle Technology*. 1st edition. Midland Information Technology and Publishing, 2002. URL: http://www.particles.org.uk.

[6] G. V. Barbosa-Canovas et al. *Food Powders: Physical Properties, Processing and Functionality*. Plenum, New York, 2005. DOI: 10.1007/0-387-27613-0. URL: https://www.springer.com/gp/book/9780306478062.

[7] P. A. Webb. "Volume and density determinations for particle technologists". In: *Micromeritics Instrument Corp.* (2001). URL: http://www.micromeritics.com.

[8] Cement Science. *What is density? how to distinguish different density definitions?* URL: https://www.cementscience.com/2013/03/what-is-density-how-to-distinguish-different-density-definitions.html.

[9] L. Svarovsky and British Materials Handling Board. *Powder Testing Guide: Methods of measuring the physical properties of bulk powders*. 1st edition. Elsevier Netherlands, 1987. URL: https://www.springer.com/gp/book/9781851661374.

[10] A. Santomaso, P. Lazzaro, and P. Canu. "Powder flowability and density ratios: the impact of granules packing". In: *Chemical Engineering Science* 58.13 (2003), pp. 2857–2874. ISSN: 0009-2509. DOI: https://doi.org/10.1016/S0009-2509(03)00137-4. URL: http://www.sciencedirect.com/science/article/pii/S0009250903001374.

[11] L. Poloni. "Density-Induced Segregation in Particles Flows". MSc Thesis, Università di Padova, 2019.

[12] A.W. Jenike. *Gravity Flow of Bulk Solids*. Bulletin of the University of Utah, 1961. URL: https://collections.lib.utah.edu/details?id=709033.

[13] G. Cavalli et al. "A shear cell study on oral and inhalation grade lactose powders". In: *Powder Technology* 372 (2020), pp. 117–127. ISSN: 0032-5910. DOI: https://doi.org/10.1016/j.powtec.2020.05.041. URL: http://www.sciencedirect.com/science/article/pii/S0032591020304046.

[14] A. Zakhvatayeva, C. Hare, and C.Y. Wu. "Suction filling of pharmaceutical powders". In: *Powder Technology* 355 (2019), pp. 438–448. ISSN: 0032-5910. DOI: https://doi.org/10.1016/j.powtec.2019.07.073. URL: http://www.sciencedirect.com/science/article/pii/S0032591019305650.

[15] Shayne Cox Gad. *Pharmaceutical Manufacturing Handobook - Production and Processes*. John Wiley & Sons, Inc., 2008. DOI: 10.1002/9780470259818. URL: https://onlinelibrary.wiley.com.

[16] Xue Tang et al. "Flow behaviour of pharmaceutical powders during rotary die filling with a paddle feeder". In: *International Journal of Pharmaceutics* 585 (2020), p. 119547. ISSN: 0378-5173. DOI: https://doi.org/10.1016/j.ijpharm.2020.119547. URL: http://www.sciencedirect.com/science/article/pii/S0378517320305317.

[17] L.C.R Schneider, I.C. Sinka, and A.C.F. Cocks. "Characterisation of the flow behaviour of pharmaceutical powders using a model die–shoe filling system". In: *Powder Technology* 173.1 (2007), pp. 59–71. ISSN: 0032-5910.

DOI: https://doi.org/10.1016/j.powtec.2006.11.015. URL: http://www.sciencedirect.com/science/article/pii/S0032591006004943.

[18] Alan J. Izenman. *Modern Mutivariate Statistical Techniques*. 2nd edition. Springer, New York, NY, 2008. DOI: 10.1007/978-0-387-78189-1. URL: https://link.springer.com/book/10.1007/978-0-387-78189-1#toc.

[19] Gareth James et al. *An Introduction to Statistical Learning*. Springer, New York, NY, 2012. DOI: 10.1007/9781461471387. URL: https://link.springer.com/book/10.1007/978-1-4614-7138-7#toc.

[20] V. K. Ohja et al. "Predictive modeling of die filling of the pharmaceutical granules using the flexible neural tree". In: *Neural Comput. & Appl.* 29 (2018), pp. 891–921. DOI: 10.1007/s00521-016-2545-8.

[21] Miroslav Kubat. *An introduction to Machine Learning*. 2nd edition. Springer, New York, NY, 2017. DOI: 10.1007/978-3-319-63913-0. URL: https://doi.org/10.1007/978-3-319-63913-0.

[22] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning - Data Mining, Inference and Prediction*. 2nd edition. Springer, New York, NY, 2009. URL: https://doi.org/10.1007/978-0-387-84858-7.

[23] Yuehui Chen et al. "Time-series forecasting using flexible neural tree model". In: *Information Sciences* 174.3 (2005), pp. 219–235. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2004.10.005. URL: http://www.sciencedirect.com/science/article/pii/S0020025504002944.

[24] V. K. Ojha, A. Abraham, and V. Snášel. "Ensemble of heterogeneous flexible neural trees using multiobjective genetic programming". In: *Applied Soft Computing* 52 (2017), pp. 909–924. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2016.09.035. URL: http://www.sciencedirect.com/science/article/pii/S156849461630494X.

[25] H. M. Zawbaa et al. "Computational intelligence modelling of pharmaceutical tabletting processes using bio-inspired optimization algorithm". In: *Advance Powder Technology* 29.12 (2018), pp. 2966–2977. URL: https://doi.org/10.1016/j.apt.2018.11.008.

[26] Mengshan Li et al. "Neural network modeling based double-population chaotic accelerated particle swarm optimization and diffusion theory for solubility prediction". In: *Chemical Engineering Research and Design* 155 (2020), pp. 98–107. ISSN: 0263-8762. DOI: https://doi.org/10.1016/j.cherd.2020.01.003. URL: http://www.sciencedirect.com/science/article/pii/S0263876220300046.

[27] L. Breiman et al. *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984. ISBN: 9780412048418. URL: https://books.google.it/books?id=JwQx-WOmSyQC.

[28] Leo Breiman. "Random Forests". In: *Machine Learning* 45 (2001), pp. 5–32. DOI: 10.1023/A:1010933404324. URL: https://link.springer.com/article/10.1023/A:1010933404324.

[29] Yong Yin et al. *Data Mining*. Springer-Verlag London, 2011. ISBN: 9781849963381. DOI: 10.1007/9781849963381. URL: https://www.springer.com/gp/book/9781849963374.

[30] A. Zakhvatayeva et al. "An experimental study of die filling of pharmaceutical powders using a rotary die filling system". In: *International Journal of Pharmaceutics* 553.1 (2018), pp. 84–96. ISSN: 0378-5173. DOI: https://doi.org/10.1016/j.ijpharm.2018.09.067. URL: http://www.sciencedirect.com/science/article/pii/S0378517318307270.

[31] R. Freeman and X. Fu. "Characterisation of power bulk, dynamic flow and shear properties in relation to die filling". In: *Powder Metallurgy* 51 (Sept. 2008), pp. 196–201. DOI: 10.1179/174329008X324115.

[32] C.-Y. Wu and A.C.F. Cocks. "Flow behaviour of powders during die filling". In: *Powder Metallurgy* 47.2 (2004), pp. 127–136. DOI: 10.1179/003258904225015617. URL: https://doi.org/10.1179/003258904225015617.

[33] Jianyi Zhang et al. "The application of terahertz pulsed imaging in characterising density distribution of roll-compacted ribbons". In: *European Journal of Pharmaceutics and Biopharmaceutics* 106 (2016). 7th International Granulation Workshop: Granulation across the Length Scales, pp. 20–25. ISSN: 0939-6411. DOI: https://doi.org/10.1016/j.ejpb.2016.01.012. URL: http://www.sciencedirect.com/science/article/pii/S0939641116000369.

[34] D. H. Wolpert and W. G. Macready. "No free lunch theorems for optimization". In: *IEEE Transactions on Evolutionary Computation* 1.1 (1997), pp. 67–82. ISSN: 1941-0026. DOI: 10.1109/4235.585893.

[35] Juan Rodríguez, Ludmila Kuncheva, and Carlos Alonso. "Rotation Forest: A New Classifier Ensemble Method". In: *IEEE transactions on pattern analysis and machine intelligence* 28 (Nov. 2006), pp. 1619–30. DOI: 10.1109/TPAMI.2006.211.

[36] Elisabeth Peeters et al. "Reduction of tablet weight variability by optimizing paddle speed in the forced feeder of a high-speed rotary tablet press". In: *Drug development and industrial pharmacy* 41.4 (2015), pp. 530–539. ISSN: 0363-9045. DOI: 10.3109/03639045.2014.884121. URL: https://doi.org/10.3109/03639045.2014.884121.

Acknowledgements

This work would have never been possible without the Erasmus+ exchange project. Therefore, I want to thank a lot the *European Commission* for making this amazing experience still possible for thousands of students around Europe. Despite the Brexit outcome, I was utterly and warmly accepted by all English students and staff. During the whole research period I really felt welcomed and part of the new student environment, so a lot of gratitude must be showed to the University of Surrey and to its personnel.
Besides, I am very grateful to prof. Charley Wu, who has always followed me step by step both in the preliminary introduction weeks and during the lockdown. His kindness, availability and fellowship played a key role during the pandemic for not giving up, staying put and focusing on my plan. Thanks to him, I was able to change immediately the initial laboratory-based research, whose fulfillment was not possible when the University was shut down. I will never forget his help and his sensitivity. Thank you Charley.
For sure, I also send an enormous thank to doc. Varun Kumar Ojha. Thanks to him, I was able to approach Machine Learning for the first time; without his generosity and patience this thesis would have been less likely to be completed on time. In addition to that, I must also underline his kindness for sharing with me his FNT software for data learning.
I am very thankful to prof. Andrea Claudio Santomaso, who followed from Italy this work and asked me frequent updates about my current situation. His care and listening were a big boost for living in a calmer way the lockdown.
Last but not least, a final acknowledgement must be directed to my family, to my University peers, to all the professors who built my professionalism, to my friends, to my landlord Emily and to all the people who I shared my path with and who shaped little by little my personality.