British Journal of Pharmacy

https://doi.org/10.5920/bjpharm.1385

British Journal of Pharmacy

www.bjpharm.hud.ac.uk

Proceedings of the 14th APS International PharmSci 2023

Application of Machine Learning for Predicting Bulk Behaviour of Active Pharmaceutical Ingredients

Martin Strachon[a]*, Marek Schongut[a]

[a]Drug Product Design, Pfizer R&D, Sandwich, Kent, United Kingdom

ARTICLE INFO

Received: 21/07/2023
Accepted: 08/08/2023
Published: 30/12/2023

*Corresponding author.
Tel.: +44 7865 631618
E-mail:
martin.strachon@pfizer.com

KEYWORDS: API; Powder; Machine Learning; Modelling.

SUMMARY

The aim of this study was to develop models for predicting powder bulk behaviour from particle properties using machine learning methods. The data consisted of various measurements of particle size, shape, and bulk properties for different active pharmaceutical ingredients. Python libraries were used to pre-process the data, select input features, and train. The models were evaluated using leave-one-out cross-validation and r^2 scores. The results showed that the models could predict the flow function coefficient (FFC), bulk density, porosity, and tap density with moderate to high accuracy. However, the models exhibited low prediction accuracy for FT-4 rheometer descriptors. The study demonstrated the feasibility and limitations of using machine learning for powder bulk behaviour prediction.

INTRODUCTION

Developing models for prediction of powder bulk behaviour such as Bulk & Tap densities and flow represented by the Flow Function Coefficient (FFC) from primary particle properties such as particle size distributions (PSD) and shape distributions (PShD) is beneficial for formulation and process development. Especially in the early development stages and particle diversity studies, when only a limited amount of material is available and therefore predominantly only particle properties are being characterized as there is not enough sample quantity for traditional bulk characterisation methods.

MATERIALS AND METHODS

Dataset containing powder PSD descriptors determined by laser diffraction (Sympatec HELOS), PSD and PShD descriptors (sphericity, convexity, aspect ratio) determined by dynamic image analysis (Sympatec QICPIC), and also results from variety of other methods such as Bulk & Tap density (Copley JV2000), Ring Shear Test (Schultze RST-XS), and FT-4 Rheometer (Freeman Technology) was built by collating available Pfizer in house data, containing approximately 2500 lots of 44 active pharmaceutical ingredients (API) many of which however could not be used due to missing relevant data.

To work with the dataset and build predictive models *Python* was employed utilising the *NumPy* (Harris, C.R. *et al.)*, *Pandas* (McKinney, W., 2010) and *scikit-learn* (Pedregosa *et al.*, 2011) libraries. Inputs were transformed with natural logarithm and all data was rescaled using the *RobustScaler*. Inputs for each model were selected based on mutual information with the response variable which was determined using Scikit-learn's *mutual_info_regression*. The inputs with the highest mutual information were used for training. Machine learning models utilised in this study were *RandomForestRegressor* and *MLPRegressor* (Multi-layer Perceptron) since, in the initial model selection, these models generally outperformed various other regressors (Linear, Partial Least Squares, Support Vector). The hyper parameters of each model were

https://doi.org/10.5920/bjpharm.1385

optimised through *RandomisedSearchCV* where a grid of possible hyper parameters was predefined. Random combination of the hyper parameters was selected, and the model was scored through a 10-fold cross validation. The best scoring set of hyper parameters was selected for training of the final model.

Leave-one-out cross-validation was used to evaluate all models. Each model was trained on all data except one sample and the response was predicted for that sample. This was repeated for each sample. Linear regression between the experimental (x_i) and predicted (y_i) values was computed to quantify accuracy using the coefficient of determination (r^2).

$$r^2 = \frac{\sum_i (x_i - y_i)^2}{\sum_i (x_i - \bar{x})^2}$$

RESULTS AND DISCUSSION

Machine learning (ML) models were trained for the following responses - Ring Shear Test Flow Function Coefficient (FFC), Bulk Density, Tap Density, Porosity and five FT-4 rheometer descriptors – Flow Rate Index (FRI), Specific Energy (SE), Compact Bulk Density (CBD), and Basic Flow Energy (BFE) and Stability Index (SI).

Table 1. *Summary of the trained models and their leave-one-out cross-validation score*

Response	Score	N input features	unique APIs	total samples
ln(FFC)	0.83	7	9	85
FFC	0.71	7	9	85
Bulk Density	0.71	25	11	81
Porosity	0.71	25	11	89
Tap Density	0.58	25	11	89
FT-4 FRI	0.47	25	6	46
FT-4 SE	0.47	15	6	46
FT-4 CBD	0.46	15	6	46
FT-4 BFE	0.34	15	6	46
FT-4 SI	< 0[a]	15	6	46

[a] only negative values of the r^2 value were achieved showing poor predictions of the models

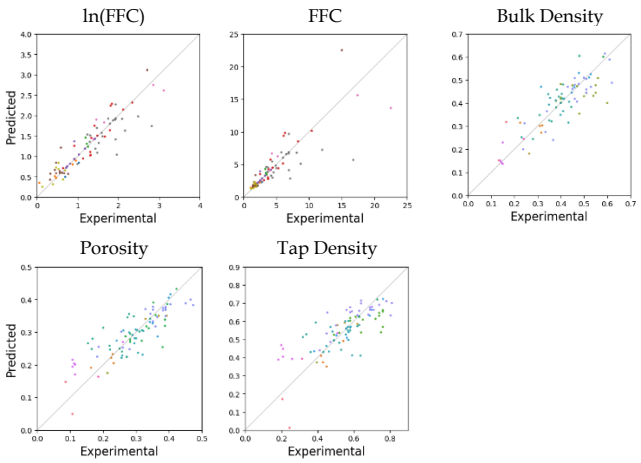

Fig. 1. *Prediction vs experimental values (models with r^2 > 0.5). Each colour corresponds to a single API*

CONCLUSIONS

Prediction models for powder bulk properties were developed. The predictions of the Flow Function Coefficient, Bulk Density and Porosity form powder particle size and shape descriptors were relatively accurate with r^2 > 0.7 and confirmed a strong relationship between the particle and powder bulk properties. Tap Density model achieved a lower accuracy with r^2 of 0.58, suggesting size and shape descriptors were insufficient to capture the packing ability of powders and perhaps other inter-particle interactions are more relevant in densely packed powders. The predictability of FT-4 rheometer descriptors was low with r^2 < 0.5, which may have been due to a small dataset of samples (n = 46), variability in the measurements, or due to the descriptors relying on a more complex powder behaviour which was not explained by the size and shape descriptors.

ACKNOWLEDGEMENTS

The author would like to acknowledge Marek Schongut and Emma Hawking for their guidance, and support.

REFERENCES

Pedregosa, F. *et al.*, 2011. Scikit-learn: Machine Learning in Python. JMLR, 12, 2825-2830

Harris, C.R., Millman, K.J., van der Walt, S.J. *et al.*, 2020. Array programming with NumPy. Nature, 585, 357–362

McKinney, W., 2010. Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference, 56-61