Utrecht University

Identifying Key Predictors of Firm Performance: An Analysis Using Machine Learning Models

Master's Thesis: Applied Data Science
INFOMTADS

Sajad Ahmadi Jozdani
5328381

Project Supervisor: Dr. Yolanda Grift
Second Reader: Dr. Linda Keijzer

July 2024

Table of Contents

Figures

Tables

Abstract

This study explores the importance of worker bodies in combination with 67 other features on firm performance using the data from the European Company Survey (ECS) 2019 dataset. The scope of this study is limited to the Germanic cluster of countries, including Austria, the Netherlands, and Germany. Firm performance was measured based on a subjective variable rated by the management of the establishments based on their profit-making situation. The main research question of the study is “What are the most influential factors on firm performance?”, and the sub-question is “How important is the role of worker bodies in predicting firm performance?”.

We used Random Forest, LightGBM, and XGBoost models using both classification and regression approaches to find the feature importance and SHAP values of the features. The results showed that worker body existence is the least important factor across all other features, while changes in production level, employment status, and motivation of employees are the most important features. At a higher level, firm characteristics, skill and training factors demonstrated the highest level of importance, whereas collaboration and external factors like product market strategy had the lowest importance values. This study is of value to econometricians and management researchers as it gives them an integrated and holistic overview of multiple features while focusing on a subset of them in their fields of interest.

Keywords: random forest, LightGBM, XGBoost, firm performance, feature importance, SHAP values, ECS2019

Introduction

Worker representative bodies initially emerged to protect workers’ rights using an intuitional approach(Hobsbawm, 1967). Throughout time, the effect of worker bodies on firm performance has been extensively analyzed across different contexts and scopes. Some researchers found a positive effect(Müller-Jentsch, 1995) while others found an adverse effect (Brunello, 1992). Irrespective of the direction of the effect, a larger question that comes to mind is the “importance” of this factor, especially when compared to other factors.

ECS is a series of extensive surveys run across European companies that opened the doors to answer this question in a systematic way. This survey initially ran in 2004 and was also implemented in 2009, 2013 and 2019. Its comprehensive underlying framework encompasses various aspects, including worker bodies and indirect employment participation, which enables us to compare the effect of different factors on firm performance.

This study used the ECS 2019 survey dataset with supervised learning methods using random forest, LightGBM, and XGBoost models. The main innovation of this study is in its integrated view, which uses various models and methods to reach reliable results across different methods. The application of both the classification and regression models, in combination with feature importance and SHAP values, resulted in more robust results.

Literature review

Worker representative bodies and firm performance

Worker representative bodies emerged during the Industrial Revolution in the late 18th and early 19th centuries. In the UK, trade unions expanded rapidly from 1889 to 1891 to three-quarters of a million participants. This trend continued, and workers joined different representative bodies. At the end of the First World War, the trade unions in the UK had a population of around 8 million workers (Hobsbawm, 1967). In 1930's, the national labor policy allowed the American industrial society to form a collective strength and develop worker bodies to protect the interest of employee against employers (Blumrosen, 1962).

In many definitions, work councils are considered an institutional representative body that represents the interests of the employees in a company to the management. They also help develop industrial and societal democracy within the firm. As an assumption, the higher participation of workers aids in involving employees in reorganization processes. As a result, it might increase commitment and, ultimately, the firm's economic efficiency (Nienhüser, 2020).

Regarding the effect of work bodies on firm performance, there are conflicting results discussed by the researchers. In one hand, Frick and Sadowski (1995) compared the job market in the USA and Germany. In their proposed framework, they argue that although many policy advisors think Europeans should deregulate their job market to grow like the USA, the main influential factor in growth is the type and degree of regulations. According to their estimates, union density does not affect turnover rates in firms. It shows that works councils and unions are complementary rather than competing institutions. Their work also shows that the presence of work councils in companies decreases employee turnover, which results in a lower loss of human capital. Additionally, Müller-Jentsch (1995) criticized the previous studies that evaluated the effect of work councils on firm performance and proposed to use of objective measures of performance instead of subjective measures. He used the firm's capital stock to measure performance and found a positive effect of work councils on profits.

On the other hand, Brunello (1992) found that Japanese unions in their sample of 979 firms reduced both the productivity and profitability of the firms, as well as regular wages. The effects were smaller in small and medium-sized firms. Dugardin (2024) used fixed-effects regression and showed that profitability decreases when firms get unionized. Firm profitability also decreases further when a second labor union emerges.

Although various studies tried to find the effect of worker bodies on firm performance, this single factor does not provide an exhaustive overview towards predicting firm performance. Other researchers went further and tried to measure the effect of different factors on firm performance. Addison and Teixeira (2024) analyzed the data of the employee and management questionnaires of the European Survey of 2013 and found that higher worker commitment (shown by employee motivation, retention, and absenteeism propensities) results in higher firm performance. They used profitability as a measure of firm performance.

Based on the approach of Addison and Teixeira (2024), we decided to consider a more exhaustive set of features and went beyond the effect of only one factor on firm performance.

Random Forest

Heath and Salzberg initially proposed random forests in 1993(Heath et al., 1993). Then, Breiman (2001) completely explained this method as a combination of tree predictors in his book. He explained that the generalization error in this method relies on the strength of each individual tree. By using a combination of trees instead of a single tree, we might be able to enhance accuracy and decrease overfitting. He argued that when the data becomes more complex with many predictors, an aggregation of decision trees provides better results in comparison to Single-Tree CART models like Decision Trees. The output of these trees is aggregated based on voting (for classification problems) or average (for regression problems) to a result. Additionally, one of the main benefits of Random Forest is its capability to learn about non-linear relationships between the predictors(Rigatti, 2017).

We used Random Forest as our base-line model. The reason to choose Random Forest was due to its widespread application in solving classification and regression problems (Shaik & Srinivasan, 2019).

XGBoost

XGBoost, short for eXtreme Gradient Boosting, was initially introduced by Chen and Guestrin (2016) as a scalable tree-boosting system. It has become a well-known model because of its robust performance and flexibility. In XGBoost, there is a sequence of models, and each model tries to correct the remaining error by the previous tree in the previous step. Initially, the data is used to train a simple model, as the first model; Then, the second model uses the output of the first model and tries to decrease the error of the previous model. This chain of intaking the output of the previous model continues up to reaching a result with the lowest degree of error.

One of the main advantages of XGBoost is its ability to handle missing values internally by treating them as an independent and separate category of observations. It makes the model more convenient to work with since real-world datasets usually contain missing values. Additionally, it supports parallel and distributed computing, which allows to analyze large datasets faster(Mitchell et al., 2018).

LightGBM

LightGBM was proposed by Microsoft Research as an effort to develop a highly efficient and scalable gradient-boosting model. It was introduced as a solution to solve the efficiency and scalability of previous models like XGBoost. In this model, instead of scanning all the data to estimate the information gain of nodes, they used a sample of data to estimate it(named as GOSS method) and combined mutually exclusive features(named as EFB method) to reduce the number of features(Ke et al., 2017).

LightGBM and XGBoost are widely compared to each other in terms of accuracy and speed. For many public datasets, LightGBM has shown a higher speed and accuracy, while for smaller datasets its advantage becomes less. Li et al. (2024) tested these two models against a variety of datasets with various parameters and found out that the leaf-wise strategy used in LigthGBM outperforms XGBoost’s layer-wise strategy.

Feature importance and SHAP values

Feature importance and SHAP values are both secondary results of some machine learning models like Random Forest, XGBoost and LightGBM. These models can show the influence and importance of each individual feature in the outcome variable. Feature importance values are widely used due their simplicity Johnsen et al. (2023), while SHAP values are a technique derived from game theory to explain the predictions of machine learning models. SHAP values indicate both the direction and the magnitude of each feature's impact on the outcome variable (Meng et al., 2020).

Framework

For this study, we used the framework proposed by Pap et al. (2022) to select and organize different factors that affect firm performance into groups, named as "Factor area". Each area consists of multiple detailed features. For example, Employee voice, which is a factor area, is made up of worker bodies existence, collective agreements, participation of workers in managerial decisions, and some other features.

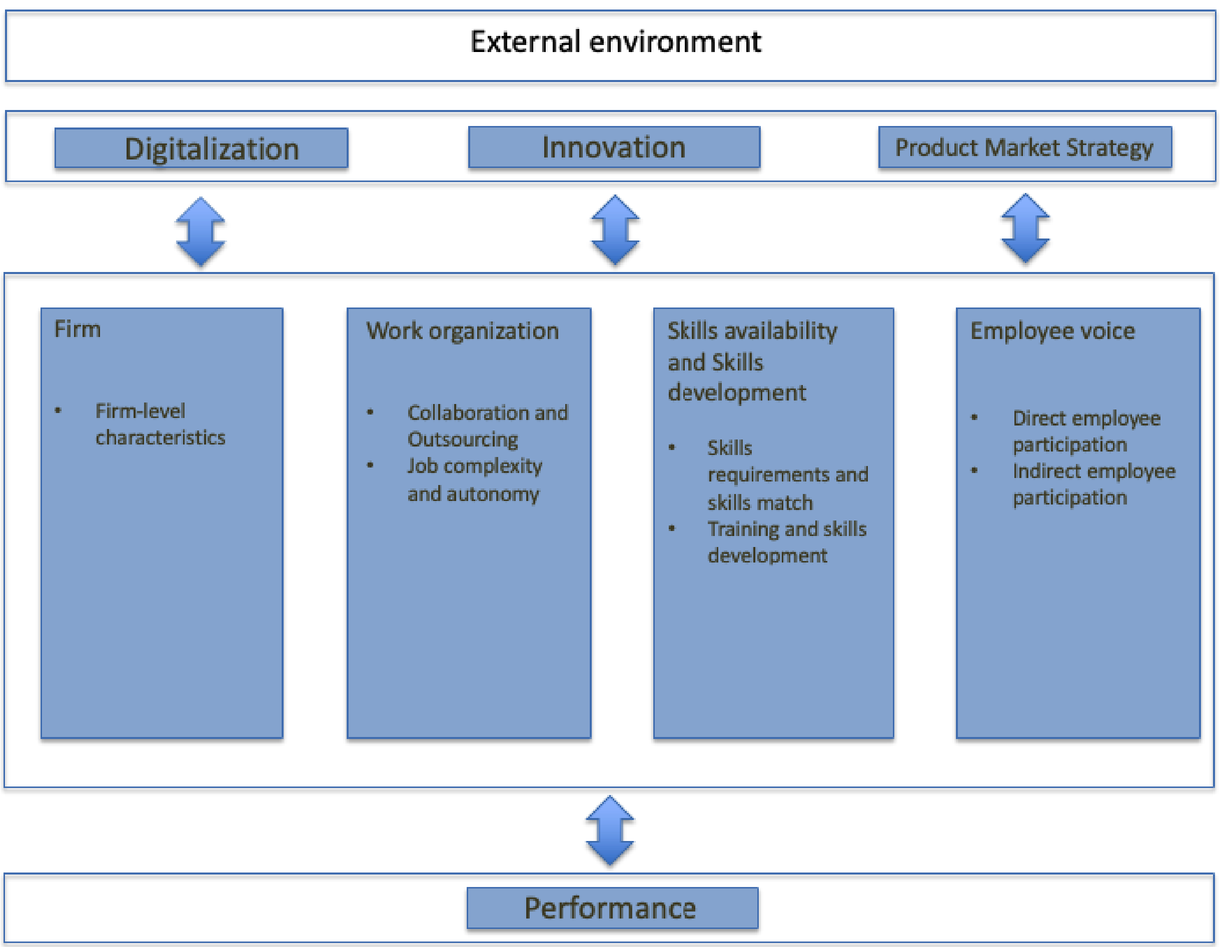


Figure 1: Theoretical Framework

In addition to the underlying framework, we considered the features mentioned by van Den Berg et al. (2013) as the firm characteristic features. These are mainly firm-level features like industry category, production level, size of the company and ....
Table 1 shows the variables used in this study and their corresponding definitions.

Table 1: Variables of the study

Main Areas	Factor Area	Explanation
External environment	Innovation	Determines the extent to which this organization is a pioneer in innovation.
	Digitalization	Indicates the degree of digitization of tasks and the processes of carrying them out.
	Product market strategy	Specifies what strategies the organization uses.
Firm	Firm Characteristics	Indicates firm-level characteristics
Work organization	Collaboration and outsourcing	Indicates the extent to which an organization uses outsourcing to carry out its activities.
	Job complexity and autonomy	Indicates the authority of the employees of that organization.
Skills availability and Skills development	Skill requirements and skill match	Explains the extent to which employees' skills match the skills required by the job.
	Training and skill development	Represents the training opportunities of the organization for employee development.
Employee voice	Direct employee participation	Determines the extent to which employees are able to express their needs directly.
	Indirect employee participation	Explains the extent to which employees indirectly express their voice.
Outputs	Firm Performance	Indicates the performance of the organization

The ECS is the first European establishment survey using push-to-web technology. It was implemented in two steps: First, a telephone screener detected the eligibility of participants for both the manager and employee surveys. Then, the eligible and selected participants received an online form containing the questionnaire. The questions used to assess each variable of the ECS 2019 framework and included in this study are detailed in Table A1 of the Appendix.

In the next chapters, we first discuss the dataset used to perform the analysis. Then, we will explain the steps regarding data cleaning, feature transformation and model training in the Methods section. In the Results section, we aggregate the results and show the effect of each factor on firm performance. Finally, in the Discussions section, we compare the results with previous studies in this field and mention the limitations and future studies.

Data and Methods

Data

In this study, we used the data from the European Company Survey (ECS) 2019. ECS is a nationwide survey of 27 EU members and the United Kingdom, run by Cedefop and Eurofound. In the ECS 2019, which is the fourth version of the survey, the information was collected from 21,869 human resource managers and 3,073 employee representatives. The respondents answer questions regarding workplace strategies, human resource management practices, employee participation, digitalization, and some other internal and external factors about the establishments they work in (CEDEFOP, 2023).

As mentioned, human resource managers and employee representatives have different questionnaires and datasets. In this study, we used the data from the managers questionnaire. It is also worth mentioning that 98% of the establishments in this survey were SMEs.

Methods

Data Cleaning and Feature Engineering

The original dataset consists of 21869 rows and 385 columns. Initially, the establishments that were non-profit or had no reported profit were removed from the profit column because it was the outcome variable. As a result, 1789 rows were removed. Then, the countries in the Germanic cluster were selected (van Den Berg et al. ,2013). This cluster consists of Austria, Germany, and the Netherlands in the country column. Thus, 2534 establishments were chosen. Table 2 shows the distribution of establishments across these 3 countries:

Table 2: The distribution of establishments across the Germanic cluster

Country	Count	proportion(rounded 2 digits)
Netherlands	967	38%
Austria	934	37%
Germany	633	25%

** Calculations based on ECS 2019 dataset*

The proportion of missing values across the dataset is high. To address this issue, we merged some features. As an example, the questions “mmerconfirm_v4_9” and “mmerconfirm_v3_9” from the questionnaire were merged to determine whether a worker body exists or not. Both questions asked about worker body existence, but respondents were able to only see one version of these two questions. As a result, in the output of the questionnaire, all values for the other version became Null values by default, in a systematic manner.

Additionally, questions about wages set by a collective agreement at the national level, sectoral level, and regional level were merged as wages set by an external party. The rest of the answers to this question (i.e., wages set at the company level, on behalf of employees, and other methods) were categorized as wages set by an internal party. Afterward, those with both types (internal and external parties) were categorized as “both types”. These features were transformed to Boolean type.

Furthermore, questions regarding skill level (skillmatch_d, overskill_d, underskill_d) were values between 0 and 1, but originally stored as string. So, we decided to convert them to float data type. These values were finally used without scaling in the final models since they were already in [0,1] range.

Except for six features, the rest of the features were all in categorical data type. We applied one-hot encoding to these features before using them in our models(Seger, 2018). One-hot encoding is a method used to transform categorical variables into separated groups of Boolean variables so that machine learning algorithms can run operations on them. We used scikit-learn’s one-hot encoder module with sparse_output parameter set to False to do the process.
For the classification models, we used scikit-learn’s label encoder module to transform the outcome variable from categorical to integer (Jia & Zhang, 2021). On the other hand, for the

regression models, we mapped the outcome categories on an ordinal scale using a mapper dictionary. The mapper dictionary assigned -1 when the outcome value was 'we made loss', 0 when it was 'we broke even', and +1 when it was 'we made profit'.

Finally, the one-hot encoded features, which were stored in a different dataframe, were joined with Boolean and float variables to form a unified feature dataframe.

Model Selection and Training

The initial outcome variable, profit, is a categorical variable with three values: "we made profit", "we broke even", "we made loss". So, we can model it as a classification task. Meanwhile, the outcome variable could be considered as a categorical ordered variable since we can assign making loss a value of -1, broke even as zero, and making profit as 1. As a result, we can model our problem as a regression task with outcome values of -1, 0, and 1.

Each approach has its own pros and cons. Addressing the problem as a classification task gives us a better understanding of the model's performance metrics, like accuracy score. For instance, we can explicitly understand that the model was able to predict 80% of the results correctly.

On the other hand, using a regression model helps us to find out the direction of the effect of the features (positive effect, negative effect, neural) on the outcome variable using SHAP values. SHAP values are a technique derived from game theory to explain the predictions of machine learning models. These values indicate both the direction and the magnitude of each feature's impact on the outcome variable.

As a result, we tested different models using classification and regression tasks with various hyper-parameters. Then, in each task, the best models that showed similar performance metrics were chosen and averaged out to find feature importance value for each feature. The averaging out of different model outputs, also called the voting method, was already used to solve various problems using machine learning (Waterschoot et al., 2022). Then, we compared the results of both tasks and reported the results.

Classification models

We used Random Forest, XGBoost, and LightGBM models and tested them across various parameters. The reason to choose Random Forest was due to its widespread application in solving classification and regression problems (Shaik & Srinivasan, 2019). Also, XGBoost has shown superior performance in comparison to ensemble methods like random forest in various benchmarking practices (Didavi et al., 2021). Additionally, LightGBM has shown faster and higher performance in large datasets in comparison to XGBoost in benchmarks by (Li et al., 2024).

After testing the algorithms with different parameters using a 5-fold cross-validation approach, the resulting accuracy scores were 0.792, 0.781, and 0.792 in order for XGBoost, LightGBM, and Random Forest. K-fold cross validation is a statistical approach in which every time a portion of the data is used to train the model and the rest is used to test the model's performance. This approach results in higher reliability for the performance scores. One of the most conventional approaches to running k-fold cross-validation is the 5-fold method. In this method, in each iteration, 80% of the data is used to train the model, and the remaining 20% is used to test the model. The accuracy score is measured as the number of correct predictions by all predictions (Sokolova et al., 2006). Although these scores are not high, we decided to adhere to the underlying framework of the study, which limited our flexibility in choosing between the features and removing some questions that decreased the models' performance metrics. Additionally, we decided not to use techniques like null imputation (Zhang, 2008) and balancing the data (Ramyachitra & Manikandan, 2014) to keep our results comparable with previous related studies that were done on the same dataset in the economics field.

Initially, we used the "feature_importances" attribute of the models to export each feature's importance(contribution) to the models' predictions. As mentioned in the data cleaning step, most features were categorical and were one-hot encoded. So, we needed to aggregate each feature's importance by summing its encoded values. For example, "prodvol_it has increased" and "prodvol_it has decreased" were aggregated to "prodvol" and their individual importance values were summed up.

As we observed, the performance of different models was close. So, we decided to take the average of the models with the highest-performing parameters to increase our results' reliability. This process is similar to the study by Johnsen et al. (2023). In their study on genotype data from the UK Biobank, they ran various ensemble-based models and averaged the feature importance scores across them to better understand which features consistently contributed to the predictions. This approach helps to identify stable and reliable features and reduce the bias that may arise from using a single model.

In Table 3, you can see the output of the classifier models:

Table 3: Feature importance output of the classification models in predicting the outcome variable (profit) and their averaged values across the features ordered by vote average importance score

Feature	importance_random_forst	importance_lgbm	importance_xgboost	vote_avg_importance_score
prodvol	0.0275	0.1732	0.1699	0.1235
chempfut	0.0210	0.1527	0.1868	0.1201
paidtraind	0.0216	0.0668	0.1462	0.0782
smainactd	0.0313	0.0694	0.0894	0.0634
lowmot	0.0134	0.0469	0.0779	0.0461
trinn	0.0137	0.0068	0.0711	0.0305
skillsmatchd	0.0546	0.0243	0.0000	0.0263
contrd	0.0198	0.0217	0.0364	0.0259
trmot	0.0138	0.0081	0.0509	0.0243
ictcompd	0.0210	0.0223	0.0276	0.0236
overskilld	0.0489	0.0216	0.0000	0.0235
wpsupp	0.0100	0.0209	0.0362	0.0223
innoprod	0.0108	0.0110	0.0427	0.0215
trski	0.0136	0.0126	0.0281	0.0181
underskilld	0.0437	0.0070	0.0000	0.0169

Feature	importance_random_forst	importance_lgbm	importance_xgboost	vote_avg_importance_score
ertrus	0.0081	0.0016	0.0367	0.0155
innomark	0.0122	0.0300	0.0000	0.0141
compprobsd	0.0219	0.0161	0.0000	0.0127
qwprel	0.0135	0.0244	0.0000	0.0126
learnnoneedd	0.0216	0.0135	0.0000	0.0117
mmepintrain	0.0172	0.0162	0.0000	0.0111
mmepindism	0.0174	0.0139	0.0000	0.0104
pmstratnps	0.0151	0.0149	0.0000	0.0100
training	0.0136	0.0144	0.0000	0.0093
regmee	0.0101	0.0175	0.0000	0.0092
onjobd	0.0220	0.0052	0.0000	0.0091
mmerinpay	0.0110	0.0160	0.0000	0.0090
comorgd	0.0211	0.0041	0.0000	0.0084
emporg	0.0086	0.0152	0.0000	0.0079
eratt	0.0087	0.0150	0.0000	0.0079
mmepintime	0.0199	0.0033	0.0000	0.0077
pcwkmachd	0.0181	0.0042	0.0000	0.0074
mmerintime	0.0091	0.0120	0.0000	0.0070
innoproc	0.0103	0.0100	0.0000	0.0068
eicomp	0.0161	0.0034	0.0000	0.0065
pmstratbq	0.0146	0.0044	0.0000	0.0064
retainemp	0.0121	0.0063	0.0000	0.0061
mmepinorg	0.0163	0.0016	0.0000	0.0060
trflex	0.0141	0.0038	0.0000	0.0060
pmstratlp	0.0142	0.0036	0.0000	0.0059
mmerintrain	0.0084	0.0092	0.0000	0.0058
sickleave	0.0098	0.0076	0.0000	0.0058
mmepinpay	0.0164	0.0009	0.0000	0.0058
dissinf	0.0125	0.0047	0.0000	0.0057
estsize	0.0093	0.0075	0.0000	0.0056
skillch	0.0123	0.0040	0.0000	0.0054
staffme	0.0112	0.0049	0.0000	0.0054
eidelay	0.0147	0.0000	0.0000	0.0049
itperfmonuse	0.0109	0.0035	0.0000	0.0048
pmstartcust	0.0132	0.0011	0.0000	0.0048
wagessetexternal	0.0078	0.0054	0.0000	0.0044
tauton	0.0114	0.0010	0.0000	0.0041
ictapp	0.0115	0.0006	0.0000	0.0040

Feature	importance_random_forst	importance_lgbm	importance_xgboost	vote_avg_importance_score
teasin	0.0102	0.0017	0.0000	0.0040
indir	0.0103	0.0012	0.0000	0.0038
teamex	0.0062	0.0044	0.0000	0.0035
mmerinorg	0.0091	0.0010	0.0000	0.0034
wagessetinternal	0.0074	0.0026	0.0000	0.0033
supchek	0.0082	0.0017	0.0000	0.0033
mmerindism	0.0094	0.0000	0.0000	0.0031
actdede	0.0090	0.0000	0.0000	0.0030
itprodimp	0.0088	0.0000	0.0000	0.0029
somedi	0.0081	0.0004	0.0000	0.0028
actprod	0.0085	0.0000	0.0000	0.0028
wagessetboth	0.0067	0.0011	0.0000	0.0026
ictrob	0.0068	0.0000	0.0000	0.0023
itperfmon	0.0064	0.0000	0.0000	0.0021
body	0.0042	0.0000	0.0000	0.0014

** Calculations based on ECS 2019 dataset*
** Blue numbers: highest values. Grey numbers: body existence feature values*

Based on the table, the random forest model has assigned importance values bigger than zero to all features. Also, the LightGBM model has assigned importance values to 60 features out of 68 features, whereas the XGBoost model has incorporated only 13 features. This behavior is due to the reason that Random Forest uses an independent tree-building process while XGBoost undertakes a sequential and regularized approach. This approach tends to be more selective, as it only addresses the remaining errors from the previous trees.

In the random forest model, the skillmatch, overskill, and underskill are the most important features in order, while in the XGBoost and LightGBM models, the most important features are employment situation(chempfut) and change in production level(prodvol). This shows that Random Forest has assigned higher importance values to features in the skill area, while XGBoost and LightGBM considered firm characteristic features to be the most important ones.

On the other hand, irrespective of the models, 'body' has the lowest importance values across all features and models. This consistency in results is the backbone of this research and what we looked for. The inherent design of machine learning models might differ a lot, but when they show highly similar results, we achieve more reliable conclusions. Regarding 'body' existence, the only model that has assigned a value other than zero to it is Random Forest, while the other models assigned a value of zero to this feature. On average, the importance of this feature across all three models is 0.14%.

Regression Models

For the regression task, we built and tested various models using 5-fold cross-validation and based on mean squared error (MSE). The average MSE values for the XGBoost, LightGBM, and Random Forest models were 0.322, 0.312, and 0.333, respectively. As it seems, none of the models could outperform the other since the MSE values are so close to each. other. As a result, we took the average of all three models across each feature as the final value representing feature importance.

As discussed earlier, our rationale for approaching the problem using regression models, while we had approached it using classifiers, was to increase the reliability of our results and find the direction of the features' effects on the outcome variable, similar to the work by Nabipour et al. (2020), and Barnes et al. (2021).

The table below shows the regression models' outputs:

Table 4: Feature importance output of the regression models in predicting the outcome variable (profit) and their averaged values across the features ordered by average importance score

Feature	importance_random_forst	importance_lgbm	importance_xgboost	avg_importance_score
chempfut	0.0414	0.2458	0.2136	0.1669
prodvol	0.0406	0.2357	0.2188	0.1650
lowmot	0.0172	0.0795	0.0878	0.0615
paidtraind	0.0285	0.0639	0.0785	0.0569
smainactd	0.0363	0.0378	0.0739	0.0493
trinn	0.0149	0.0235	0.0666	0.0350
pmstratnps	0.0143	0.0294	0.0510	0.0315
compprobsd	0.0238	0.0066	0.0615	0.0306
skillsmatchd	0.0723	0.0108	0.0000	0.0277
estsize	0.0081	0.0210	0.0492	0.0261
overskilld	0.0626	0.0100	0.0000	0.0242
training	0.0118	0.0105	0.0485	0.0236
underskilld	0.0620	0.0073	0.0000	0.0231
sickleave	0.0068	0.0071	0.0507	0.0215
qwprel	0.0133	0.0389	0.0000	0.0174
trski	0.0185	0.0158	0.0000	0.0114
onjobd	0.0181	0.0138	0.0000	0.0106
learnnoneedd	0.0217	0.0100	0.0000	0.0105
innomark	0.0113	0.0199	0.0000	0.0104
contrd	0.0188	0.0095	0.0000	0.0095
mmerinpay	0.0100	0.0154	0.0000	0.0085
innoproc	0.0112	0.0140	0.0000	0.0084
mmepindism	0.0151	0.0083	0.0000	0.0078
wpsupp	0.0091	0.0135	0.0000	0.0075
mmepintrain	0.0181	0.0040	0.0000	0.0074
comorgd	0.0204	0.0015	0.0000	0.0073
mmepintime	0.0180	0.0037	0.0000	0.0072

Feature	importance_random_forst	importance_lgbm	importance_xgboost	avg_importance_score
ictcompd	0.0202	0.0000	0.0000	0.0067
retainemp	0.0091	0.0089	0.0000	0.0060
pcwkmachd	0.0180	0.0000	0.0000	0.0060
itperfmonuse	0.0110	0.0057	0.0000	0.0056
pmstratbq	0.0115	0.0044	0.0000	0.0053
trflex	0.0120	0.0026	0.0000	0.0049
mmepinorg	0.0142	0.0000	0.0000	0.0047
staffme	0.0123	0.0018	0.0000	0.0047
eidelay	0.0110	0.0023	0.0000	0.0044
mmepinpay	0.0130	0.0000	0.0000	0.0043
eicomp	0.0128	0.0000	0.0000	0.0043
pmstratlp	0.0121	0.0000	0.0000	0.0040
ictapp	0.0084	0.0036	0.0000	0.0040
trmot	0.0115	0.0000	0.0000	0.0038
skillch	0.0098	0.0016	0.0000	0.0038
mmerintime	0.0114	0.0000	0.0000	0.0038
pmstartcust	0.0112	0.0000	0.0000	0.0037
mmerintrain	0.0081	0.0030	0.0000	0.0037
tauton	0.0108	0.0000	0.0000	0.0036
dissinf	0.0090	0.0019	0.0000	0.0036
eratt	0.0080	0.0022	0.0000	0.0034
teasin	0.0093	0.0000	0.0000	0.0031
mmerindism	0.0091	0.0000	0.0000	0.0030
ictrob	0.0052	0.0034	0.0000	0.0029
actprod	0.0076	0.0000	0.0000	0.0025
indir	0.0076	0.0000	0.0000	0.0025
ertrus	0.0075	0.0000	0.0000	0.0025
regmee	0.0072	0.0000	0.0000	0.0024
innoprod	0.0068	0.0000	0.0000	0.0023
supchek	0.0063	0.0000	0.0000	0.0021
itprodimp	0.0062	0.0000	0.0000	0.0021
somedi	0.0061	0.0000	0.0000	0.0020
mmerinorg	0.0057	0.0000	0.0000	0.0019
wagessetexternal	0.0041	0.0013	0.0000	0.0018
actdede	0.0047	0.0000	0.0000	0.0016
emporg	0.0045	0.0000	0.0000	0.0015
itperfmon	0.0040	0.0000	0.0000	0.0013
wagessetboth	0.0030	0.0000	0.0000	0.0010

Feature	importance_random_forst	importance_lgbm	importance_xgboost	avg_importance_score
wagessetinternal	0.0022	0.0000	0.0000	0.0007
teamex	0.0021	0.0000	0.0000	0.0007
body	0.0011	0.0000	0.0000	0.0004

* *Calculations based on ECS 2019 dataset*
* *Blue numbers: highest values. Grey numbers: body existence feature values*

Regarding the results table, the random forest model used all features in its predictions, so no feature has a value of zero. On the other hand, the LigthtGBM model has assigned feature importance values to 40 features, while XGBoost used only nine features. We observed almost the same behavior in the previous results table, but this time the XGBoost and LightGBM were stricter.

In addition, like the previous table, the most important features in the random forest model are “skillmatch”, “overskill”, and “underskill”. For the XGBoost and LightGBM models, the most important features are production change and employment situation(“prodvol” and “chempfut”). Also, the least important feature is ‘body’, with a value of zero in the XGBoost and LightGBM models.

In addition to feature importance values, we analyzed the regression models' SHAP values to compare them with previous results. SHAP(Shapely) values are a method for explaining the output of machine learning models based on game theory models. They can show the influence of each feature on the outcome variable and their importance. Variables that get a negative sign, tend to decrease the model’s outcome variable towards negative values, while values that get positive signs help to increase the outcome variable of the model towards higher values. Table 5 shows the aggregated SHAP values based on the regression models and their average importance values across these models (Meng et al., 2020).

Table 5: SHAP values of the regression models in predicting the outcome variable (profit) and their averaged values across the features ordered by average value

Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_xgb	Mean_SHAP_Value_lgb	avg_value
prodvol	0.1694	0.2933	0.1139	0.1922
chempfut	0.0363	0.4261	0.0089	0.1571
smainactd	0.0341	0.1594	0.1034	0.0990
paidtraind	0.0138	0.0352	0.0659	0.0383
estsize	0.0078	0.0278	0.0609	0.0322
skillsmatchd	0.0777	0.0000	0.0122	0.0300
overskilld	0.0765	0.0000	0.0111	0.0292
underskilld	0.0674	0.0000	0.0164	0.0279
innoproc	0.0067	0.0000	0.0744	0.0270
trinn	0.0141	0.0317	0.0192	0.0216
skillch	0.0042	0.0000	0.0531	0.0191
wpsupp	0.0086	0.0000	0.0479	0.0188
qwprel	0.0086	0.0000	0.0436	0.0174

Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_xgb	Mean_SHAP_Value_lgb	avg_value
trski	0.0111	0.0000	0.0406	0.0172
pmstratnps	0.0101	0.0050	0.0351	0.0167
retainemp	0.0063	0.0000	0.0419	0.0161
itperfmonuse	0.0082	0.0000	0.0346	0.0143
compprobsd	0.0189	0.0103	0.0116	0.0136
mmepintrain	0.0133	0.0000	0.0269	0.0134
mmerinpay	0.0219	0.0000	0.0168	0.0129
innomark	0.0090	0.0000	0.0252	0.0114
lowmot	0.0091	0.0002	0.0207	0.0100
mmerintime	0.0273	0.0000	0.0000	0.0091
mmerintrain	0.0145	0.0000	0.0115	0.0087
training	0.0050	0.0094	0.0096	0.0080
contrd	0.0179	0.0000	0.0049	0.0076
onjobd	0.0134	0.0000	0.0087	0.0074
learnnoneedd	0.0123	0.0000	0.0094	0.0073
dissinf	0.0095	0.0000	0.0111	0.0069
pmstratbq	0.0052	0.0000	0.0137	0.0063
mmepintime	0.0163	0.0000	0.0020	0.0061
mmepinorg	0.0173	0.0000	0.0000	0.0058
comorgd	0.0157	0.0000	0.0014	0.0057
ictcompd	0.0161	0.0000	0.0000	0.0054
eratt	0.0130	0.0000	0.0029	0.0053
ictrob	0.0082	0.0000	0.0064	0.0049
pcwkmachd	0.0145	0.0000	0.0000	0.0048
mmepindism	0.0056	0.0000	0.0070	0.0042
trflex	0.0043	0.0000	0.0081	0.0041
ictapp	0.0078	0.0000	0.0042	0.0040
eidelay	0.0041	0.0000	0.0074	0.0038
sickleave	0.0039	0.0016	0.0054	0.0036
mmerinorg	0.0102	0.0000	0.0000	0.0034
trmot	0.0092	0.0000	0.0000	0.0031
staffme	0.0068	0.0000	0.0016	0.0028
supchek	0.0076	0.0000	0.0000	0.0025
pmstartcust	0.0071	0.0000	0.0000	0.0024
innoprod	0.0071	0.0000	0.0000	0.0024
actprod	0.0063	0.0000	0.0000	0.0021
indir	0.0063	0.0000	0.0000	0.0021
somedi	0.0063	0.0000	0.0000	0.0021

Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_xgb	Mean_SHAP_Value_lgb	avg_value
mmerindism	0.0062	0.0000	0.0000	0.0021
tauton	0.0061	0.0000	0.0000	0.0020
pmstratlp	0.0059	0.0000	0.0000	0.0020
wagessetexternal	0.0051	0.0000	0.0003	0.0018
mmepinpay	0.0049	0.0000	0.0000	0.0016
itprodimp	0.0044	0.0000	0.0000	0.0015
teasin	0.0044	0.0000	0.0000	0.0015
wagessetboth	0.0043	0.0000	0.0000	0.0014
emporg	0.0042	0.0000	0.0000	0.0014
eicomp	0.0041	0.0000	0.0000	0.0014
itperfmon	0.0037	0.0000	0.0000	0.0012
ertrus	0.0036	0.0000	0.0000	0.0012
regmee	0.0035	0.0000	0.0000	0.0012
actdede	0.0031	0.0000	0.0000	0.0010
teamex	0.0021	0.0000	0.0000	0.0007
wagessetinternal	0.0016	0.0000	0.0000	0.0005
body	0.0010	0.0000	0.0000	0.0003

** Calculations based on ECS 2019 dataset*
** Blue numbers: highest values. Grey numbers: body existence feature values*

The SHAP values also showed similar behavior to the feature importance values in the regression model since they both used regression models as their basis. Speaking of the random forest model, production volume change(“prodvol”) became the most important feature in contrast to the two previous result tables, but in the XGBoost model, employment situation(“chempfut”) was the most important feature. Similar to previous results, ‘body’ existence showed the lowest importance value in comparison to other features.

In the next step, following the study's underlying framework, we aggregated the feature importance values for each “factor” area.

Results

In this part, we explain the models’ results and analyze them further. First, we discuss the model outputs shown in the previous tables, and then we show the aggregated data across each ‘factor’ area.

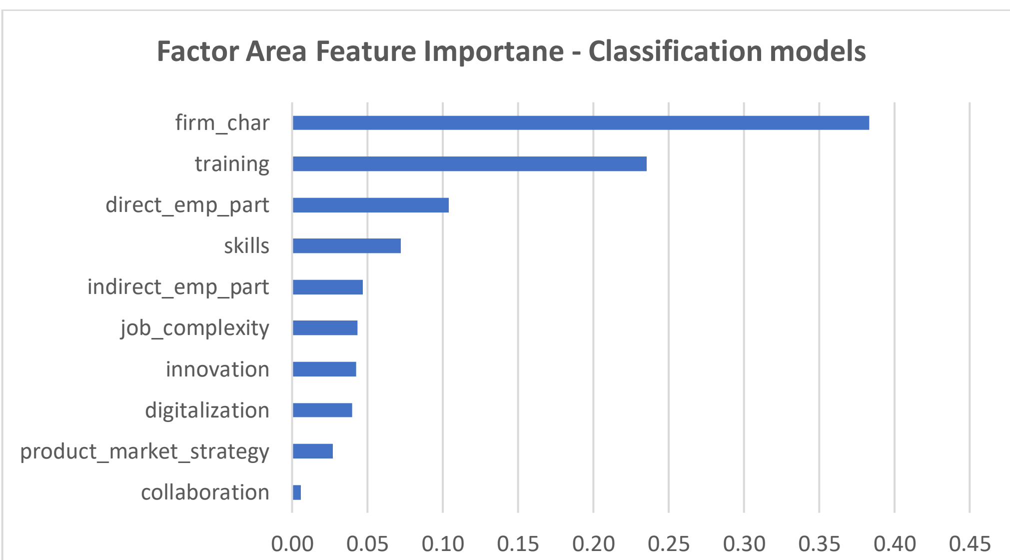
We see almost the same results across all three tables regarding the ‘body’ feature, which shows worker body existence. It has the lowest effect on firm performance in comparison to all other features across all the outputs from the classification and regression models. In fact, the only algorithm that assigned a contribution to this feature was the random forest. Since the importance values are normalized, it shows that in the regression models, ‘body existence’ only has around 0.04% importance. Also, in the SHAP values, this feature has a value of 0.03% contribution. In the classification models, it has about 0.1% importance. As a result, all model outputs insist on the low importance of this feature in comparison to other features.

The importance value of the classification models and the SHAP values of the regression models show that production level change (prodvol) is the most important variable in predicting an establishment's performance. In order, these measures assigned values of around 12.3% and 19.2% contribution to this feature. Additionally, in the regression models, this feature is the second most important feature, with a value of 16.5%, just below the most important feature.

At a higher level of aggregation and regrading 'factor' area, all three importance measures assigned the highest value to the 'firm characteristic' factor. The firm characteristics feature has values of about 38%, 51%, and 53% in the classification feature importance, regression feature importance, and regression SHAP measures. Also, 'training' is the second most important feature among the three measures, with values of around 23%, 17%, and 13%, respectively, for the classification importance values, regression importance values, and regression SHAP values.

Figure 2 shows the aggregated feature importance value for each factor using the classification models. The 'firm_char' feature, which shows 'firm characteristic' related features, has the highest impact with a value of around 38%. The second most important factor is 'training.' It shows the different aspects of training employees, like on-the-job training, paid training, and the opportunities to learn from experienced colleagues. The least important factor is 'collaboration,' with a value of almost 0.06%. This feature shows the engagement of the establishment in production, design, and outsourcing processes.

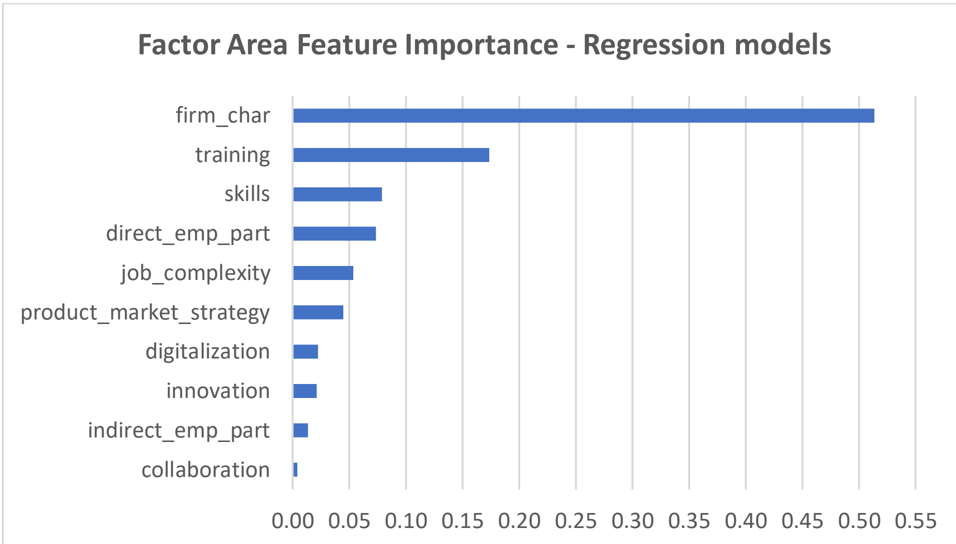
Figure2: Factor Area Feature Importance - Classification models



* *Calculations based on ECS 2019 dataset*
* *firm_char consists of features like production level, industry, size of the company, and …*
* *Indirect_emp_part includes features like worker body existence, collective agreements, and …*

Figure 3 shows the feature importance for each factor area based on the regression models. Like the classification models, firm characteristics and training areas have the highest impact, and collaboration has the lowest impact. Meanwhile, the firm characteristic factor has a higher weight, around 52%, in comparison to its value in the classification model results. In other words, the regression models assign more than half of the firm performance results only to this factor.

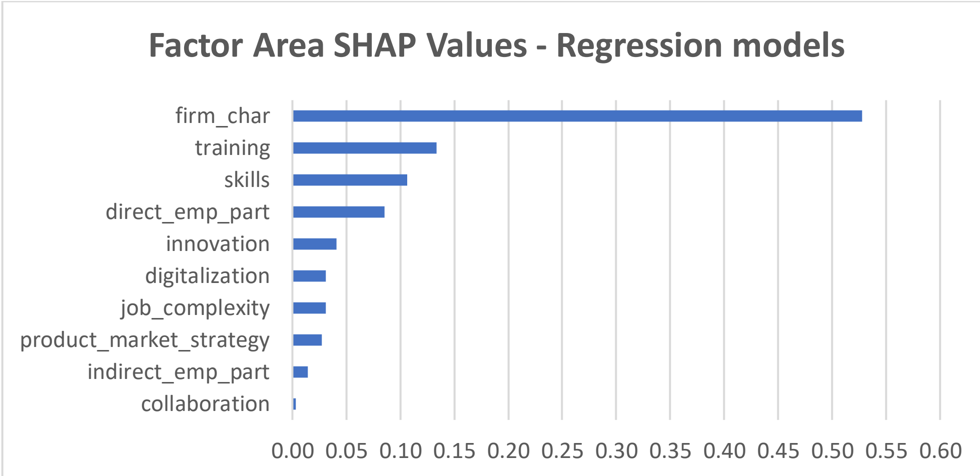
Figure 3: Factor Area Feature Importance - Regression models



* *Calculations based on ECS 2019 dataset*
* *firm_char consists of features like production level, industry, size of the company, and …*
* *Indirect_emp_part includes features like worker body existence, collective agreements, and …*

Figure 4 demonstrates the output of the regression models using SHAP values. Compared to the previous results, the firm characteristic area has a higher contribution, with a value of around 53%. Training, skills, and direct employee participation have values between about 8% and 13%. Other factors, like innovation, digitalization, and job complexity, have values of less than 5% each. The Collaboration factor in this metric has a contribution of nearly 0.3%, which is the lowest amount in comparison to the previous results. We can observe that the SHAP values distribution is more asymmetric than that of other methods.

Figure 4: Factor Area SHAP Values - Regression models



** Calculations based on ECS 2019 dataset*
** firm_char consists of features like production level, industry, size of the company, and …*
** Indirect_emp_part includes features like worker body existence, collective agreements, and …*

Table 6 shows all the results gathered in one table. In order, the firm characteristic and training factor areas have the highest values across all three methods. Notice that these results are highly aggregated and consistent across multiple models and methods so that we can make highly reliable conclusions at this level. This indicates that the firm characteristics and training factors play a crucial role in firm performance. In addition, the skills factor is in the third position of importance in the regression models' feature importance and SHAP values, while in the fourth rank for the classification models. It generally shows the importance of this factor in comparison to other factors. Other factors like direct and indirect employee participation, job complexity, and also external factors (including digitalization, innovation, and product market strategy) differ in their orders across different models.

Table 6: Feature importance and SHAP values for Factor areas

factor area	classification feature importance	regression feature importance	regression SHAP values
firm_char	0.3833	0.5138	0.5275
training	0.2354	0.1739	0.1334
direct_emp_part	0.1039	0.0738	0.0853
skills	0.0722	0.0788	0.1062
indirect_emp_part	**0.0468**	**0.0138**	**0.0141**
job_complexity	0.0434	0.0535	0.0309
innovation	0.0423	0.0211	0.0408
digitalization	0.0398	0.0226	0.0312
product_market_strategy	0.0271	0.0446	0.0274
collaboration	0.0058	0.0041	0.0031

** Calculations based on ECS 2019 dataset*
** Worker body existence is part of the Indirect_emp_part factor*

As discussed earlier, one of the reasons for measuring SHAP values was to find the direction of the effect of the features. The details of the SHAP values are available in Table A2 in the Appendix. The "body" feature has two values, one showing the existence of a worker body and the second one showing the absence of a worker body. Due to the one-hot encoding process, these values are used and reported as separate features in the models. The SHAP value for the existence of a body is -3.74E-5, and for the absence of a body is -1.68E-5. Both values are negative, showing a lowering effect on firm performance. The less negative SHAP value for the absence of a body suggests a smaller negative impact compared to the existence of a body. This indicates that while both states negatively impact firm performance, the absence of a worker body has a less severe negative impact than the existence of a worker body.

Discussion

In a similar study by Pap et al. (2022) on the same dataset and using the same features, except for the firm characteristic features, the researchers found that 'collaboration' and 'job complexity' are the most important factor areas. However, in the current study, with some changes in the underlying framework and methods, these features didn't appear to have a high importance value. In this study, Job complexity is mostly ranked in the middle of other factor areas. In contrast, collaboration, which is considered the most important factor in that study, is the least important factor across all three methods. These differences might be related to differences in the firm performance(outcome) variable, methods, and features. Pap et al. (2022) used a genetic algorithm to select independent variables and then used the BART method as their machine-learning model and their firm outcome variable was the both employee well-being and firm performance.

On the other hand, 'Indirect employee participation', which is the least important factor in their study, also received low importance scores in our study. This factor includes the 'worker body' feature. Thus, in both studies, a low importance value is assigned to this factor. Additionally, the results of both studies are aligned regarding external variables, including innovation, digitalization, and product market strategy. In both studies, the most important factors are Innovation, Product Market Strategy, and Digitalization, respectively.

The results of this study might be used by researchers in the fields of economics, corporate governance, and management. This study provides an integrated overview of 68 different attributes in one of the most widely used surveys across European firms. One of the main contributions of this study is to help researchers determine the most important control variables when measuring the effect of a single feature or a group of features on the outcome variable. Firm characteristic features in this study would be suitable candidates as control variables for future studies by econometricians.

Also, the importance of skills and training was shown almost consistently across the models' outputs. This provides ideas to researchers in corporate governance and management to dig deeper into the sub-features of these two factors and compare them against different outcome variables.

Regarding the limitations of this study and future studies, we undertook a strict approach to adhere to the underlying framework of this study. We also made minimal changes to the original features to make them comparable with previous studies in other fields like economics and management. As a result, we didn’t use various available methods in machine learning, such as up-sampling the outcome variables and null values imputation, to keep the distribution of the data as untouched as possible. Also, considering 68 different features due to following the framework limited our flexibility in the feature selection step. In fact, we took a top-down approach to training our models, starting with a thorough framework and making small changes within the framework.

Future studies might take a bottom-up approach, i.e., starting without a framework and choosing only the best features that help increase the models’ performance metrics. In addition, using the mentioned methods, like up-sampling and null imputation, might help increase the performance of the models. Furthermore, as Müller-Jentsch (1995) proposed, it is recommended to use objective measures to evaluate firm performance instead of subjective measures rated by managers to reduce bias in the outcomes. Regarding worker body importance, it is worthwhile to notice that the scope of this study was limited to Germanic cluster countries, and the ECS 2019 dataset is comprised of mostly SME firms. Future studies are recommended to use other categories of the countries of the same dataset or other datasets that are better representatives of firms with different sizes.

References

Addison, J. T., & Teixeira, P. (2024). Worker commitment and establishment performance in Europe. *The Manchester School*, *92*(1), 40-66.

Barnes, D., Polanco, L., & Perea, J. A. (2021). A comparative study of machine learning methods for persistence diagrams. *Frontiers in Artificial Intelligence*, *4*, 681174.

Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5-32.

Brunello, G. (1992). The effect of unions on firm performance in Japanese manufacturing. *ILR Review*, *45*(3), 471-487.

CEDEFOP. (2023). *European Company Survey (ECS) 2019 | CEDEFOP*. https://www.cedefop.europa.eu/en/landing-page/european-company-survey-ecs-2019

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining,

Didavi, A. B., Agbokpanzo, R. G., & Agbomahena, M. (2021). Comparative study of Decision Tree, Random Forest and XGBoost performance in forecasting the power output of a photovoltaic system. 2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART),

Dugardin, F.-A. (2024). Labor Unions and Firms Performance: A Panel Analysis. *Available at SSRN 4702115*.

Frick, B., & Sadowski, D. (1995). Works councils, unions, and firm performance. *Institutional Frameworks and Labor Market Performance', Routledge, London and New York*, 46-81.

Heath, D., Kasif, S., & Salzberg, S. (1993). k-DT: A multi-tree learning method. Proc. of the Second Int. Workshop on Multistrategy Learning,

Hobsbawm, E. (1967). Trade union history. *The economic history review*, *20*(2), 358-364.

Jia, B.-B., & Zhang, M.-L. (2021). Multi-dimensional classification via sparse label encoding. International Conference on Machine Learning,

Johnsen, P. V., Strümke, I., Langaas, M., DeWan, A. T., & Riemer-Sørensen, S. (2023). Inferring feature importance with uncertainties with application to large genotype data. *PLOS Computational Biology*, *19*(3), e1010963.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.

Li, S., Dong, X., Ma, D., Dang, B., Zang, H., & Gong, Y. (2024). Utilizing the LightGBM Algorithm for Operator User Credit Assessment Research. *arXiv preprint arXiv:2403.14483*.

Meng, Y., Yang, N., Qian, Z., & Zhang, G. (2020). What makes an online review more helpful: an interpretation framework using XGBoost and SHAP values. *Journal of Theoretical and Applied Electronic Commerce Research*, *16*(3), 466-490.

Mitchell, R., Adinets, A., Rao, T., & Frank, E. (2018). Xgboost: Scalable GPU accelerated learning. *arXiv preprint arXiv:1806.11248*.

Müller-Jentsch, W. (1995). Germany: From collective voice to co-management. In *Works councils: Consultation, representation, and cooperation in industrial relations* (pp. 53-78). University of Chicago Press.

Nabipour, M., Nayyeri, P., Jabani, H., Shahab, S., & Mosavi, A. (2020). Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis. *IEEE access*, *8*, 150199-150212.

Nienhüser, W. (2020). Works councils. In *Handbook of research on employee voice* (pp. 259-276). Edward Elgar Publishing.

Pap, J., Mako, C., Illessy, M., Kis, N., & Mosavi, A. (2022). Modeling organizational performance with machine learning. *Journal of Open Innovation: Technology, Market, and Complexity*, *8*(4), 177.

Ramyachitra, D., & Manikandan, P. (2014). Imbalanced dataset classification and solutions: a review. *International Journal of Computing and Business Research (IJCBR)*, *5*(4), 1-29.

Rigatti, S. J. (2017). Random forest. *Journal of Insurance Medicine*, *47*(1), 31-39.

Seger, C. (2018). An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing. In.

van Den Berg, A., Grift, Y., van Witteloostuijn, A., Boone, C., & Van der Brempt, O. (2013). The effect of employee workplace representation on firm performance: A cross-country comparison within Europe.

Shaik, A. B., & Srinivasan, S. (2019). A brief survey on random forest ensembles in classification model. International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 2,

Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006). Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. Australasian joint conference on artificial intelligence,

Waterschoot, C., van den Hemel, E., & van den Bosch, A. (2022). Detecting minority arguments for mutual understanding: A moderation tool for the online climate change debate. Proceedings of the 29th International Conference on Computational Linguistics,

Zhang, S. (2008). Parimputation: From imputation and null-imputation to partially imputation. *IEEE Intell. Informatics Bull.*, *9*(1), 32-38.

Appendix

Table A1: The questions used to assess each variable of the ECS 2019 framework

Factor	Question code	Question
firm characteristics	prodvol	Since 2016, how has the amount of goods or services produced by this establishment changed
	est_size	Establishment size in number of employees
	smainact_d	What is the correct main sector of activity?
	sickleave	Do you think the level of sickness leave in this establishment is too high?
	retainemp	How difficult is it for this establishment to retain employees?
	lowmot	Overall, how motivated do you think employees in this establishment are?
	qwprel	How would you describe the relations between management and employees in this establishment in general?
	chempfut	In the next three years, how do you expect the total number of employees in this establishment to change?
Collaboration and outsourcing	actprod	Is this establishment engaged in the production of goods, assembly of parts or delivery of services?
	actdede	Is this establishment engaged in the design or development of new products or services?
Job complexity and autonomy	teamex	A team is a group of people working together with a shared responsibility for the execution of allocated tasks. Team members can come from the same unit or from different units across the establishment. Do you have any teams fitting this definition in this establishment?
	teasin	With regard to the employees doing teamwork, do most of them work in a single team or do most of them work in more than one team?
	tauton	Please think about the tasks to be performed by these teams. Who usually decides how the tasks are distributed within the team?
	supchek	Different establishments use different approaches to manage the way employees carry out their tasks. Which of these two statements best describes the general approach to management at this establishment? Please think about the approach that is used the most by managers.
	compprobs_d	For how many employees in this establishment does their job include finding solutions to unfamiliar problems they are confronted with? Your best estimate is good enough
	comorg_d	For how many employees in this establishment does their job include independently organising their own time and scheduling their own tasks? Your best estimate is good enough.
	pcwkmach_d	For how many employees at this establishment is the pace of work determined by machines or computers? Your best estimate is good enough.
Skills requirements and skills match	skillsmatch_d	What percentage of employees have the skills that are about right to do the job?
	overskill_d	What percentage of employees have a higher level of skills than is needed in their job?
	underskill_d	What percentage of employees have a lower level of skills than is needed in their job?
	skillch	How quickly do the knowledge and skills needed from the employees in this establishment change?
Training and skill development	contr_d	How many employees in this establishment are in jobs that require continuous training? Your best estimate is good enough
	learnnoneed_d	How many employees in this establishment are in jobs that offer limited opportunities to learn new things? Your best estimate is good enough.
	training	What are the most important ways through which employees in this establishment can become more skilled at their jobs?
	piadtrain	In 2018, how many employees in this establishment participated in training sessions on the establishment premises or at other locations during paid working time? Your best estimate is good enough
	onjob_d	In 2018, how many employees in this establishment have received on-the-job training or other forms of direct instruction in the workplace from more experienced colleagues? Your best estimate is good enough.
	wpsupp	Workload and work schedules can prevent the participation of employees in training activities. Which of the following statements best describes what happens in practice at this establishment?
	trski	How important is “Ensuring that employees have the skills they need to do their current job” for providing training to employees in this establishment?

Factor	Question code	Question
	trflex	How important is “Allowing employees to acquire skills they need to do other jobs than their current job. For instance, to allow for job rotation or career advancement.” for providing training to employees in this establishment?
	trinn	How important is “Increasing the capacity of employees to articulate ideas about improvements to the establishment” for providing training to employees in this establishment?
	trmot	How important is “Improving employee morale” for providing training to employees in this establishment?
Indirect employee participation	emporg	Is the company member of any employers' organisation which participates in the negotiation of collective agreements?
	canat	A collective agreement negotiated at the national or cross-sectoral level - wages set by
	casec	A collective agreement negotiated at the sectoral level - wages set by
	careg	A collective agreement at the regional level - wages set by
	cacom	A collective agreement negotiated at the establishment or company level - wages set by
	caocc	A collective agreement negotiated on behalf of employees with a specific occupation - wages set by
	caoth	Another type of collective agreement - wages set by
	mmerconfirm_v4_9	There is no official employee representation - official employee representation doesn't exist
	mmerconfirm_v3_9	There is no official employee representation - official employee representation exist
	eratt	How would you describe the general attitude of the employee representation at this establishment?
	indir	Management prefers to consult with the employee representation or directly with employees?
	ertus	#N/A
Direct employee participation	regmee	Meetings between employees and their immediate manager - practice used to involve employees in how work is organised
	staffme	Meetings open to all employees at the establishment - practice used to involve employees in how work is organised
	dissinf	Dissemination of information through newsletters, website, notice boards - practice used to involve employees in how work is organised
	somedi	Discussions with employees through social media or in online discussion - practice used to involve employees in how work is organised
	eidelay	To what extent does involving employees cause delays in the implementation of changes?
	eicomp	To what extent does involving employees in work organisation changes give the establishment a competitive advantage?
	mmepinorg	The organisation and efficiency of work processes - since 2016, employees directly influenced management decisions
	mmepindism	Dismissals - since 2016, employees directly influenced management decisions
	mmepintrain	Training and skill development - since 2016, employees directly influenced management decisions
	mmepintime	Working time arrangements - since 2016, employees directly influenced management decisions
	mmepinpay	Payment schemes - since 2016, employees directly influenced management decisions
	mmerinorg	The organisation and efficiency of work processes - since 2016, employee representation directly influenced management decisions
	mmerindism	Dismissals - since 2016, employee representation directly influenced management decisions
	mmerintrain	Training and skill development - since 2016, employee representation directly influenced management decisions
	mmerintime	Working time arrangements - since 2016, employee representation directly influenced management decisions
	mmerinpay	Payment schemes - since 2016, employee representation directly influenced management decisions

Factor	Question code	Question
Innovation	innoprod	Since 2016, has this establishment introduced any new or significantly changed products or services?
	innoproc	Since 2016, has this establishment introduced any new/changed processes either for producing goods or supplying services?
	innomark	Since 2016, has this establishment introduced any new or significantly changed marketing methods?
Digitalization	ictcomp_d	How many employees in this establishment use personal computers or laptops to carry out their daily tasks?
	ictapp	Since 2016, did this establishment purchase any software that was specifically developed or customised to meet the needs?
	ictrob	Robots carry complex series of actions automatically, which may include the interaction with people. Does this establishment use robots?
	itprodimp	Does this establishment use data analytics to improve the processes of production or service delivery?
	itperfmon	Does this establishment use data analytics to monitor employee performance?
	itperfmonuse	Since 2016, how would you say the use of data analytics in this establishment has changed?
Product market strategy	pmstratlp	Offering products or services at lower prices than the competition - important for the competitive success
	pmstratbq	Offering products or services that are of better quality than those offered by the competition - important for the competitive success
	pmstartcust	Customising products or services to meet specific customer requirements - important for the competitive success
	pmstratnps	Regularly developing products, services or processes that are new to the market - important for the competitive success
firm performance	profit	In 2018, did this establishment make a profit?

Table A2: The details of the SHAP values

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
prodvol_ithasincreased	prodvol	0.004531032371	-0.00033961114	-0.00040639515
prodvol_ithasdecreased	prodvol	0.001635242286	9.58E-05	-7.61E-05
chempfut_itwilldecrease	chempfut	0.001550692709	3.31E-05	0.00068879634
mmerinpay_notatall	mmerinpay	0.000796072986	-7.87E-05	0
mmerintime_skipped	mmerintime	0.0006709120012		
skillsmatchd_59.0	skillsmatchd	0.0004485297618		
underskilld_71.0	underskilld	0.0004019098166		
trski_notveryimportant	trski	0.0003619993876	-0.0001075707506	0
lowmot_notverymotivated	lowmot	0.0003298928112	-2.47E-05	-3.88E-07
qwprel_bad	qwprel	0.0003069290325	0	0
mmepintime_skipped	mmepintime	0.0002785054143		
underskilld_60.0	underskilld	0.0002770976981		
overskilld_65.0	overskilld	0.0002730706511		
ictcompd_20%to39%	ictcompd	0.0002625917146		0
overskilld_68.0	overskilld	0.000258658402		
skillsmatchd_32.0	skillsmatchd	0.0002541202548		
chempfut_itwillincrease	chempfut	0.0002378861473	-8.76E-06	-1.21E-05
smainactd_informationandcommunication	smainactd	0.0002367930354	0	0
innomark_yes,newtothemarket	innomark	0.0002350685476		0
wpsupp_workloadandworkschedulesareadjustedtoallowemployeestoparticipateintrainingandprofessionaldevelopmentacti	wpsupp	0.0002314156465	-0.0002245970444	0
trinn_veryimportant	trinn	0.000222903774	9.01E-05	5.21E-05
pmstratnps_4	pmstratnps	0.0002194910514	0	0
overskilld_55.0	overskilld	0.0002170364901		
wpsupp_participationintrainingandprofessionaldevelopmentactivitiesisonlypossibleifworkloadandworkschedulesallow	wpsupp	0.0002133490615	0	0
mmerintime_toamoderateextent	mmerintime	0.0002056376182	0	0
paidtraind_20%to39%	paidtraind	0.0002051369813		5.79E-05
trmot_fairlyimportant	trmot	0.0001963202431	0	0
pcwkmachd_lessthan20%	pcwkmachd	0.000196120281		0
underskilld_22.0	underskilld	0.0001951818959		
underskilld_24.0	underskilld	0.000191028929		
trmot_veryimportant	trmot	0.0001861351593	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
skillsmatchd_53.0	skillsmatchd	0.0001815385845		
smainactd_arts,entertainmentandrecreation	smainactd	0.0001776203057		0
supchek_skipped	supchek	0.0001754795362		
teasin_mostofthemworkinmorethanoneteam	teasin	0.0001692869002	0	0
skillsmatchd_25.0	skillsmatchd	0.0001686663245		
underskilld_10.0	underskilld	0.00016307256		
underskilld_59.0	underskilld	0.0001622389467		
pmstratnps_3	pmstratnps	0.0001582522515	-0.0001226443177	8.19E-06
paidtraind_lessthan20%	paidtraind	0.0001575324522		0
overskilld_8.0	overskilld	0.0001507091129		
itperfmonuse_ithasstayedaboutthesame	itperfmonuse	0.0001451122784	0	0
training_3	training	0.000144885702	4.51E-05	1.55E-05
itperfmonuse_skipped	itperfmonuse	0.0001445422542		
innomark_no	innomark	0.0001348739575	8.45E-05	0
skillsmatchd_61.0	skillsmatchd	0.0001310107777		
regmee_yes,onanirregularbasis	regmee	0.0001305807526		0
lowmot_fairlymotivated	lowmot	0.0001283136187	0	0
overskilld_75.0	overskilld	0.0001251992602		
retainemp_fairlydifficult	retainemp	0.0001246995713	0.0001965354046	0
itprodimp_skipped	itprodimp	0.0001197037452		
skillsmatchd_35.0	skillsmatchd	0.0001151508956		
trski_notatallimportant	trski	0.000112198074	0	0
ertrus_toamoderateextent	ertrus	0.0001082495849	0	0
underskilld_7.0	underskilld	0.000107259932		
overskilld_100.0	overskilld	0.0001026829935		
mmerinpay_toagreatextent	mmerinpay	9.75E-05	0	0
overskilld_31.0	overskilld	9.68E-05		
smainactd_miningandquarrying	smainactd	9.40E-05	0	0
skillsmatchd_76.0	skillsmatchd	9.34E-05		
skillsmatchd_44.0	skillsmatchd	9.29E-05		
pcwkmachd_noneatall	pcwkmachd	9.26E-05	0	0
onjobd_skipped	onjobd	9.15E-05		
mmepindism_toasmallextent	mmepindism	9.11E-05	0	0
underskilld_29.0	underskilld	9.10E-05		
smainactd_otherserviceactivities	smainactd	8.94E-05	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
wagessetinternal	wagessetinternal	8.88E-05	0	0
skillsmatchd_84.0	skillsmatchd	8.74E-05		
skillsmatchd_28.0	skillsmatchd	8.67E-05		
smainactd_accommodationandfoodserviceactivities	smainactd	8.52E-05	0	0
eratt_fairlyconstructive	eratt	8.52E-05	0	0
underskilld_47.0	underskilld	8.45E-05		
underskilld_14.0	underskilld	8.25E-05		
learnnoneedd_all	learnnoneedd	8.06E-05	0	0
skillsmatchd_72.0	skillsmatchd	8.06E-05		
overskilld_4.0	overskilld	8.00E-05		
overskilld_10.0	overskilld	7.89E-05		
mmepindism_toamoderateextent	mmepindism	7.86E-05	-2.53E-05	0
skillch_nochangeatall	skillch	7.74E-05	0	0
staffme_yes,onaregularbasis	staffme	7.64E-05		0
contrd_skipped	contrd	7.39E-05		
overskilld_90.0	overskilld	7.36E-05		
underskilld_2.0	underskilld	7.27E-05		
underskilld_28.0	underskilld	7.26E-05		
prodvol_skipped	prodvol	7.12E-05		
itperfmon_skipped	itperfmon	7.05E-05		
comorgd_20%to39%	comorgd	6.86E-05		0
overskilld_60.0	overskilld	6.70E-05		
tauton_skipped	tauton	6.63E-05		
learnnoneedd_60%to79%	learnnoneedd	6.63E-05		0
compprobsd_noneatall	compprobsd	6.57E-05	0	0
mmerindism_toasmallextent	mmerindism	6.56E-05	0	0
trflex_notveryimportant	trflex	6.55E-05	0	0
overskilld_18.0	overskilld	6.53E-05		
skillsmatchd_22.0	skillsmatchd	6.50E-05		
qwprel_neithergoodnorbad	qwprel	6.48E-05	0.0001421600361	0
sickleave_skipped	sickleave	6.38E-05		
underskilld_12.0	underskilld	6.34E-05		
skillch_skipped	skillch	6.25E-05		
overskilld_13.0	overskilld	6.11E-05		
mmepinorg_skipped	mmepinorg	6.09E-05		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
smainactd_electricity,gas,steamandairconditioningsupply	smainactd	6.02E-05		0
overskilld_28.0	overskilld	5.78E-05		
overskilld_40.0	overskilld	5.74E-05		
compprobsd_all	compprobsd	5.71E-05	0	0
skillsmatchd_12.0	skillsmatchd	5.71E-05		
retainemp_skipped	retainemp	5.70E-05		
underskilld_35.0	underskilld	5.50E-05		
actprod_skipped	actprod	5.43E-05		
mmepintrain_nodecisionsweremadeinthisarea	mmepintrain	5.41E-05	-6.37E-05	0
pcwkmachd_40%to59%	pcwkmachd	5.40E-05		0
skillsmatchd_78.0	skillsmatchd	5.32E-05		
actdede_yes	actdede	5.30E-05	0	0
skillsmatchd_29.0	skillsmatchd	5.27E-05		
pmstratnps_2	pmstratnps	5.07E-05	4.21E-05	0
trski_fairlyimportant	trski	5.06E-05	8.27E-05	0
skillsmatchd_87.0	skillsmatchd	5.02E-05		
eratt_notveryconstructive	eratt	4.94E-05	1.37E-05	0
overskilld_35.0	overskilld	4.88E-05		
ertrus_toagreatextent	ertrus	4.82E-05	0	0
skillsmatchd_69.0	skillsmatchd	4.78E-05		
compprobsd_skipped	compprobsd	4.74E-05		
skillsmatchd_96.0	skillsmatchd	4.50E-05		
mmepinpay_notatall	mmepinpay	4.49E-05	0	0
skillsmatchd_67.0	skillsmatchd	4.46E-05		
overskilld_38.0	overskilld	4.34E-05		
trflex_veryimportant	trflex	4.33E-05	0	0
smainactd_realestateactivities	smainactd	4.32E-05	0	0
smainactd_watersupply;sewerage,wastemanagementandremediationactivities	smainactd	4.26E-05		0
underskilld_9.0	underskilld	4.16E-05		
overskilld_67.0	overskilld	4.03E-05		
eratt_veryconstructive	eratt	3.81E-05	0	0
overskilld_27.0	overskilld	3.76E-05		
underskilld_85.0	underskilld	3.74E-05		
overskilld_45.0	overskilld	3.73E-05		
pcwkmachd_all	pcwkmachd	3.70E-05	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
skillsmatchd_40.0	skillsmatchd	3.64E-05		
overskilld_33.0	overskilld	3.53E-05		
lowmot_skipped	lowmot	3.52E-05		
pmstratbq_1	pmstratbq	3.36E-05	0	0
indir_managementprefersnottoconsultwithemployeesortheirrepresentatives	indir	3.32E-05	0	0
overskilld_25.0	overskilld	3.31E-05		
underskilld_1.0	underskilld	3.30E-05		
overskilld_24.0	overskilld	3.28E-05		
eicomp_skipped	eicomp	3.23E-05		
mmerintrain_toagreatextent	mmerintrain	3.13E-05	0	0
underskilld_42.0	underskilld	3.09E-05		
smainactd_transportationandstorage	smainactd	2.85E-05	4.92E-07	0
underskilld_38.0	underskilld	2.84E-05		
overskilld_2.0	overskilld	2.76E-05		
skillsmatchd_56.0	skillsmatchd	2.71E-05		
overskilld_42.0	overskilld	2.60E-05		
overskilld_7.0	overskilld	2.59E-05		
underskilld_23.0	underskilld	2.58E-05		
overskilld_29.0	overskilld	2.51E-05		
innomark_skipped	innomark	2.50E-05		
ictapp_yes	ictapp	2.49E-05	-8.08E-06	0
skillsmatchd_79.0	skillsmatchd	2.37E-05		
skillsmatchd_86.0	skillsmatchd	2.33E-05		
pcwkmachd_skipped	pcwkmachd	2.29E-05		
underskilld_16.0	underskilld	2.24E-05		
mmepindism_skipped	mmepindism	2.22E-05		
skillsmatchd_45.0	skillsmatchd	2.14E-05		
ertrus_notatall	ertrus	1.86E-05	0	0
overskilld_44.0	overskilld	1.82E-05		
underskilld_13.0	underskilld	1.73E-05		
trflex_skipped	trflex	1.71E-05		
pmstratbq_3	pmstratbq	1.71E-05	0	0
overskilld_11.0	overskilld	1.65E-05		
underskilld_40.0	underskilld	1.64E-05		
skillsmatchd_88.0	skillsmatchd	1.60E-05		
mmepinpay_skipped	mmepinpay	1.48E-05		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
itprodimp_no	itprodimp	1.45E-05	0	0
retainemp_notverydifficult	retainemp	1.44E-05	0	0
overskilld_53.0	overskilld	1.43E-05		
skillsmatchd_91.0	skillsmatchd	1.31E-05		
skillsmatchd_73.0	skillsmatchd	1.24E-05		
onjobd_lessthan20%	onjobd	1.21E-05		0
skillsmatchd_38.0	skillsmatchd	1.21E-05		
lowmot_notatallmotivated	lowmot	1.21E-05	0	0
skillsmatchd_33.0	skillsmatchd	1.15E-05		
training_2	training	1.14E-05	0	0
skillsmatchd_47.0	skillsmatchd	1.12E-05		
skillsmatchd_36.0	skillsmatchd	1.07E-05		
overskilld_62.0	overskilld	1.06E-05		
skillsmatchd_55.0	skillsmatchd	1.05E-05		
contrd_20%to39%	contrd	1.04E-05		0
underskilld_26.0	underskilld	1.01E-05		
skillsmatchd_89.0	skillsmatchd	1.00E-05		
training_skipped	training	9.82E-06		
overskilld_9.0	overskilld	9.66E-06		
overskilld_1.0	overskilld	9.51E-06		
skillsmatchd_8.0	skillsmatchd	8.34E-06		
underskilld_27.0	underskilld	8.14E-06		
skillsmatchd_26.0	skillsmatchd	8.07E-06		
skillsmatchd_18.0	skillsmatchd	7.48E-06		
mmerintime_toasmallextent	mmerintime	7.20E-06	0	0
mmepinorg_toamoderateextent	mmepinorg	7.04E-06	0	0
paidtraind_skipped	paidtraind	6.73E-06		
mmerindism_toamoderateextent	mmerindism	6.46E-06	0	0
underskilld_17.0	underskilld	6.23E-06		
supchek_managerscreateanenvironmentinwhichemployeescanautonomouslycarryouttheirtasks	supchek	5.92E-06	0	0
underskilld_45.0	underskilld	5.82E-06		
overskilld_3.0	overskilld	5.60E-06		
skillsmatchd_94.0	skillsmatchd	5.35E-06		
qwprel_verybad	qwprel	4.52E-06	0	0
eidelay_skipped	eidelay	4.12E-06		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
overskilld_74.0	overskilld	3.97E-06		
ictrob_skipped	ictrob	3.81E-06		
mmepindism_nodecisionsweremadeinthisarea	mmepindism	3.72E-06	0	0
underskilld_19.0	underskilld	3.69E-06		
underskilld_32.0	underskilld	3.57E-06		
ertrus_skipped	ertrus	2.28E-06		
skillsmatchd_68.0	skillsmatchd	2.15E-06		
smainactd_administrativeandsupportserviceactivities	smainactd	1.76E-06	0	0
somedi_no	somedi	8.61E-07	0	0
overskilld_17.0	overskilld	5.89E-07		
underskilld_76.0	underskilld	0		
skillsmatchd_3.0	skillsmatchd	0		
underskilld_41.0	underskilld	0		
skillsmatchd_43.0	skillsmatchd	0		
underskilld_43.0	underskilld	0		
skillsmatchd_54.0	skillsmatchd	0		
overskilld_59.0	overskilld	0		
underskilld_44.0	underskilld	0		
skillsmatchd_48.0	skillsmatchd	0		
skillsmatchd_11.0	skillsmatchd	0		
qwprel_skipped	qwprel	0		
skillsmatchd_58.0	skillsmatchd	0		
skillsmatchd_31.0	skillsmatchd	0		
underskilld_92.0	underskilld	0		
underskilld_91.0	underskilld	0		
skillsmatchd_9.0	skillsmatchd	0		
skillsmatchd_66.0	skillsmatchd	0		
skillsmatchd_17.0	skillsmatchd	0		
underskilld_90.0	underskilld	0		
underskilld_65.0	underskilld	0		
underskilld_69.0	underskilld	0		
skillsmatchd_6.0	skillsmatchd	0		
underskilld_87.0	underskilld	0		
overskilld_26.0	overskilld	0		
skillsmatchd_15.0	skillsmatchd	0		
skillsmatchd_34.0	skillsmatchd	0		
underskilld_82.0	underskilld	0		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
underskilld_72.0	underskilld	0		
underskilld_75.0	underskilld	0		
overskilld_32.0	overskilld	0		
overskilld_36.0	overskilld	0		
overskilld_34.0	overskilld	0		
overskilld_78.0	overskilld	0		
overskilld_93.0	overskilld	0		
underskilld_37.0	underskilld	0		
skillsmatchd_27.0	skillsmatchd	0		
overskilld_89.0	overskilld	0		
overskilld_86.0	overskilld	0		
overskilld_85.0	overskilld	0		
overskilld_84.0	overskilld	0		
comorgd_skipped	comorgd	0		
skillsmatchd_21.0	skillsmatchd	0		
overskilld_56.0	overskilld	0		
overskilld_95.0	overskilld	0		
overskilld_72.0	overskilld	0		
overskilld_71.0	overskilld	0		
skillsmatchd_97.0	skillsmatchd	0		
innoprod_skipped	innoprod	0		
overskilld_69.0	overskilld	0		
skillsmatchd_23.0	skillsmatchd	0		
overskilld_58.0	overskilld	0		
overskilld_66.0	overskilld	0		
overskilld_64.0	overskilld	0		
overskilld_94.0	overskilld	0		
overskilld_92.0	overskilld	0		
skillsmatchd_39.0	skillsmatchd	0		
overskilld_47.0	overskilld	0		
overskilld_39.0	overskilld	0		
overskilld_41.0	overskilld	0		
ictcompd_skipped	ictcompd	0		
overskilld_37.0	overskilld	0		
skillsmatchd_99.0	skillsmatchd	0		
underskilld_31.0	underskilld	0		
underskilld_34.0	underskilld	0		
skillsmatchd_2.0	skillsmatchd	0		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
skillsmatchd_13.0	skillsmatchd	0		
skillsmatchd_41.0	skillsmatchd	0		
skillsmatchd_98.0	skillsmatchd	0		
underskilld_100.0	underskilld	0		
underskilld_36.0	underskilld	0		
skillsmatchd_7.0	skillsmatchd	0		
overskilld_52.0	overskilld	0		
overskilld_6.0	overskilld	-2.15E-07		
underskilld_skipped	underskilld	-1.62E-06		
overskilld_23.0	overskilld	-1.83E-06		
underskilld_89.0	underskilld	-1.91E-06		
skillsmatchd_5.0	skillsmatchd	-2.10E-06		
skillsmatchd_16.0	skillsmatchd	-2.11E-06		
overskilld_16.0	overskilld	-2.46E-06		
learnnoneedd_80%to99%	learnnoneedd	-2.48E-06		0
lowmot_verymotivated	lowmot	-2.54E-06	7.24E-05	0
skillsmatchd_42.0	skillsmatchd	-3.15E-06		
overskilld_43.0	overskilld	-3.26E-06		
skillsmatchd_skipped	skillsmatchd	-3.44E-06		
skillsmatchd_19.0	skillsmatchd	-4.15E-06		
overskilld_87.0	overskilld	-4.32E-06		
pmstratnps_skipped	pmstratnps	-4.83E-06		
skillsmatchd_14.0	skillsmatchd	-5.09E-06		
overskilld_12.0	overskilld	-5.28E-06		
trflex_notatallimportant	trflex	-5.46E-06	0	0
skillsmatchd_64.0	skillsmatchd	-5.47E-06		
regmee_skipped	regmee	-5.63E-06		
skillsmatchd_71.0	skillsmatchd	-5.88E-06		
overskilld_14.0	overskilld	-5.97E-06		
mmepinpay_toasmallextent	mmepinpay	-6.35E-06	0	0
overskilld_19.0	overskilld	-6.39E-06		
overskilld_skipped	overskilld	-6.47E-06		
pmstratlp_skipped	pmstratlp	-7.04E-06		
underskilld_6.0	underskilld	-7.12E-06		
chempfut_skipped	chempfut	-7.45E-06		
staffme_skipped	staffme	-7.49E-06		
underskilld_78.0	underskilld	-7.77E-06		
skillsmatchd_92.0	skillsmatchd	-8.23E-06		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
skillsmatchd_46.0	skillsmatchd	-8.36E-06		
eicomp_notatall	eicomp	-8.37E-06	0	0
pmstartcust_skipped	pmstartcust	-8.54E-06		
overskilld_80.0	overskilld	-8.63E-06		
underskilld_80.0	underskilld	-9.00E-06		
innoprod_yes,newtothe market	innoprod	-9.09E-06		0
mmepintime_toamodera teextent	mmepintime	-9.39E-06	0	0
teamex_no	teamex	-9.90E-06	0	0
underskilld_4.0	underskilld	-1.02E-05		
overskilld_21.0	overskilld	-1.05E-05		
dissinf_skipped	dissinf	-1.06E-05		
innoproc_skipped	innoproc	-1.07E-05		
mmerindism_skipped	mmerindism	-1.08E-05		
skillsmatchd_57.0	skillsmatchd	-1.19E-05		
itperfmon_yes	itperfmon	-1.23E-05	0	0
mmepinpay_nodecisions weremadeinthisarea	mmepinpay	-1.26E-05	0	0
overskilld_57.0	overskilld	-1.28E-05		
underskilld_11.0	underskilld	-1.31E-05		
skillsmatchd_83.0	skillsmatchd	-1.52E-05		
pmstartcust_4	pmstartcust	-1.53E-05	0	0
emporg_skipped	emporg	-1.55E-05		
eicomp_toasmallextent	eicomp	-1.65E-05	0	0
body_bodydoesnotexist	body	-1.68E-05	0	0
pmstratbq_skipped	pmstratbq	-1.70E-05		
underskilld_3.0	underskilld	-1.78E-05		
mmerindism_toagreatex tent	mmerindism	-1.82E-05	0	0
skillsmatchd_0.0	skillsmatchd	-1.86E-05		
skillsmatchd_82.0	skillsmatchd	-1.91E-05		
eidelay_toamoderateext ent	eidelay	-1.92E-05	0	0
ictcompd_noneatall	ictcompd	-1.99E-05	0	0
mmepintime_toagreatex tent	mmepintime	-2.00E-05	0	0
underskilld_21.0	underskilld	-2.05E-05		
skillsmatchd_74.0	skillsmatchd	-2.13E-05		
underskilld_15.0	underskilld	-2.23E-05		
skillch_fairlyquickly	skillch	-2.32E-05	0	0
indir_managementprefe rstoconsultwiththeempl	indir	-2.32E-05	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
oyeerepresentationandwithemployeesdirectly				
teasin_mostofthemworkinasingleteam	teasin	-2.33E-05	0	0
dissinf_no	dissinf	-2.37E-05	-5.18E-05	0
pmstratlp_4	pmstratlp	-2.40E-05	0	0
regmee_no	regmee	-2.52E-05	0	0
skillsmatchd_62.0	skillsmatchd	-2.52E-05		
ertrus_toasmallextent	ertrus	-2.61E-05	0	0
itperfmonuse_ithasincreased	itperfmonuse	-2.62E-05	0	0
ictcompd_40%to59%	ictcompd	-2.83E-05		0
skillch_notveryquickly	skillch	-2.97E-05	-0.0002492266251	0
underskilld_50.0	underskilld	-2.98E-05		
sickleave_no	sickleave	-3.01E-05	0	0
qwprel_verygood	qwprel	-3.34E-05	0	0
eratt_notatallconstructive	eratt	-3.35E-05	0	0
somedi_skipped	somedi	-3.47E-05		
wpsupp_skipped	wpsupp	-3.51E-05		
regmee_yes,onaregularbasis	regmee	-3.57E-05		0
paidtraind_60%to79%	paidtraind	-3.59E-05		0
underskilld_8.0	underskilld	-3.62E-05		
trmot_skipped	trmot	-3.70E-05		
body_bodyexists	body	-3.74E-05	0	0
retainemp_verydifficult	retainemp	-3.86E-05	0	0
contrd_40%to59%	contrd	-3.94E-05		0
comorgd_60%to79%	comorgd	-3.97E-05		0
trski_skipped	trski	-4.14E-05		
mmerintrain_toasmallextent	mmerintrain	-4.17E-05	0	0
eicomp_toamoderateextent	eicomp	-4.17E-05	0	0
learnnoneedd_skipped	learnnoneedd	-4.20E-05		
skillsmatchd_50.0	skillsmatchd	-4.24E-05		
contrd_all	contrd	-4.29E-05	2.22E-05	0
skillch_veryquickly	skillch	-4.29E-05	0	0
trmot_notatallimportant	trmot	-4.48E-05	0	0
skillsmatchd_81.0	skillsmatchd	-4.69E-05		
innoproc_yes,newtotheestablishment,butnotnewtothemarket	innoproc	-4.70E-05		0
trmot_notveryimportant	trmot	-4.79E-05	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
actdede_no	actdede	-5.10E-05	0	0
teasin_skipped	teasin	-5.24E-05		
onjobd_noneatall	onjobd	-5.37E-05	0	0
mmerinorg_skipped	mmerinorg	-5.46E-05		
eidelay_notatall	eidelay	-5.53E-05	0	0
trski_veryimportant	trski	-5.66E-05	0	0
pmstartcust_2	pmstartcust	-5.67E-05	0	0
mmepindism_toagreatextent	mmepindism	-5.74E-05	0	0
comorgd_noneatall	comorgd	-5.77E-05	0	0
eidelay_toagreatextent	eidelay	-5.77E-05	0	0
onjobd_80%to99%	onjobd	-5.81E-05		0
pmstratlp_2	pmstratlp	-6.01E-05	0	0
mmepintrain_skipped	mmepintrain	-6.08E-05		
paidtraind_noneatall	paidtraind	-6.11E-05	-9.24E-05	0
mmepindism_notatall	mmepindism	-6.29E-05	-7.72E-06	0
compprobsd_lessthan20%	compprobsd	-6.52E-05		0
pcwkmachd_80%to99%	pcwkmachd	-6.64E-05		0
mmepintrain_notatall	mmepintrain	-6.68E-05	0	0
skillsmatchd_85.0	skillsmatchd	-6.70E-05		
skillsmatchd_30.0	skillsmatchd	-6.83E-05		
trinn_notveryimportant	trinn	-6.88E-05	0	0
actdede_skipped	actdede	-6.88E-05		
skillsmatchd_80.0	skillsmatchd	-6.90E-05		
mmerinpay_skipped	mmerinpay	-7.22E-05		
qwprel_good	qwprel	-7.29E-05	6.21E-05	0
learnnoneedd_20%to39%	learnnoneedd	-7.33E-05		0
estsize_10to49employees	estsize	-7.35E-05	-0.0002101551579	0
ictcompd_all	ictcompd	-7.45E-05	0	0
ictcompd_lessthan20%	ictcompd	-7.48E-05		0
learnnoneedd_lessthan20%	learnnoneedd	-7.52E-05		0
mmepinpay_toagreatextent	mmepinpay	-7.85E-05	0	0
mmerinorg_toamoderateextent	mmerinorg	-7.96E-05	0	0
indir_skipped	indir	-8.00E-05		
indir_managementpreferstoconsultwithemployeesdirectly	indir	-8.06E-05	0	0
skillsmatchd_100.0	skillsmatchd	-8.13E-05		

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
emporg_yes	emporg	-8.17E-05	0	0
tauton_tasksaredistributedbyasuperior	tauton	-8.17E-05	0	0
skillsmatchd_93.0	skillsmatchd	-8.18E-05		
somedi_yes,onaregularbasis	somedi	-8.55E-05		0
pmstartcust_1	pmstartcust	-8.77E-05	0	0
skillsmatchd_20.0	skillsmatchd	-8.94E-05		
staffme_no	staffme	-9.06E-05	0	0
underskilld_20.0	underskilld	-9.13E-05		
skillsmatchd_95.0	skillsmatchd	-9.16E-05		
eidelay_toasmallextent	eidelay	-9.34E-05	3.45E-05	0
ictrob_no	ictrob	-9.48E-05	0	0
underskilld_70.0	underskilld	-9.49E-05		
paidtraind_40%to59%	paidtraind	-9.51E-05		0
mmerinorg_toagreatextent	mmerinorg	-9.76E-05	0	0
mmerinpay_toamoderateextent	mmerinpay	-9.99E-05	0	0
onjobd_all	onjobd	-0.0001010458428	0	0
paidtraind_80%to99%	paidtraind	-0.0001040592519		0
innoprod_yes,newtotheestablishment,butnotnewtothemarket	innoprod	-0.0001064055574		0
mmerintrain_toamoderateextent	mmerintrain	-0.0001064926435	-5.40E-05	0
smainactd_construction	smainactd	-0.0001066233393	-4.15E-05	0
pmstratbq_2	pmstratbq	-0.0001069432511	-6.43E-05	0
trinn_skipped	trinn	-0.0001070896434		
teamex_yes	teamex	-0.0001078538696	0	0
paidtraind_all	paidtraind	-0.0001085146964	0	0
trflex_fairlyimportant	trflex	-0.0001094059537	-3.82E-05	0
actprod_no	actprod	-0.0001098702112	0	0
innomark_yes,newtotheestablishment,butnotnewtothemarket	innomark	-0.0001102344406		0
pmstratlp_3	pmstratlp	-0.0001109429246	0	0
compprobsd_60%to79%	compprobsd	-0.0001121179269		-1.70E-05
training_1	training	-0.0001121303565	0	0
overskilld_70.0	overskilld	-0.0001121670607		
skillsmatchd_75.0	skillsmatchd	-0.0001121759837		
itprodimp_yes	itprodimp	-0.0001138334532	0	0
mmerintime_toagreatextent	mmerintime	-0.0001141354206	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
estsize_250employeesormore	estsize	-0.0001151078824	-7.56E-05	-4.57E-05
retainemp_notatalldifficult	retainemp	-0.0001156522421	0	0
mmepinpay_toamoderateextent	mmepinpay	-0.0001157498878	0	0
pmstratbq_4	pmstratbq	-0.0001157875493	0	0
underskilld_33.0	underskilld	-0.0001165174716		
compprobsd_80%to99%	compprobsd	-0.0001227305695		0
itperfmon_no	itperfmon	-0.0001228173218	0	0
ictapp_skipped	ictapp	-0.0001233074176		
skillsmatchd_10.0	skillsmatchd	-0.0001234374176		
sickleave_yes	sickleave	-0.0001238926551	-2.53E-05	-2.64E-06
mmepintrain_toagreatextent	mmepintrain	-0.0001240407288	0	0
contrd_lessthan20%	contrd	-0.0001274089288		0
skillsmatchd_60.0	skillsmatchd	-0.0001277320521		
pmstratlp_1	pmstratlp	-0.0001291356337	0	0
pmstratnps_1	pmstratnps	-0.0001302968169	0	0
mmepinorg_notatall	mmepinorg	-0.0001317633038	0	0
eicomp_toagreatextent	eicomp	-0.0001323576719	0	0
onjobd_60%to79%	onjobd	-0.0001324699398		0
overskilld_50.0	overskilld	-0.0001354404176		
indir_managementpreferstoconsultwiththeemployeerepresentation	indir	-0.000135612783	0	0
emporg_no	emporg	-0.0001393196858	0	0
pcwkmachd_60%to79%	pcwkmachd	-0.000140214208		0
innoproc_no	innoproc	-0.0001411098751	0	0
onjobd_40%to59%	onjobd	-0.0001417175916		0
itperfmonuse_ithasdecreased	itperfmonuse	-0.0001437884808	0	0
mmepintrain_toamoderateextent	mmepintrain	-0.000150364676	0	0
underskilld_18.0	underskilld	-0.0001514559425		
skillsmatchd_90.0	skillsmatchd	-0.0001529952942		
mmepintime_notatall	mmepintime	-0.0001533743793	0	0
smainactd_manufacturing	smainactd	-0.0001549850905	0	0
comorgd_lessthan20%	comorgd	-0.0001554542867		0
mmerinpay_toasmallextent	mmerinpay	-0.0001570128778	0	0
onjobd_20%to39%	onjobd	-0.0001577423046		0
trinn_notatallimportant	trinn	-0.0001578766839	0	0
contrd_80%to99%	contrd	-0.0001590250562		0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
learnnoneedd_noneatall	learnnoneedd	-0.0001596783569	0	0
underskilld_0.0	underskilld	-0.0001608811257		
comorgd_80%to99%	comorgd	-0.0001622198227		0
skillsmatchd_70.0	skillsmatchd	-0.0001646317935		
mmerinorg_notatall	mmerinorg	-0.0001653212589	0	0
comorgd_all	comorgd	-0.0001674522956	0	0
mmepintime_toasmallextent	mmepintime	-0.0001695516475	0	0
underskilld_5.0	underskilld	-0.0001705409952		
mmerinorg_toasmallextent	mmerinorg	-0.0001720051169	0	0
skillsmatchd_77.0	skillsmatchd	-0.0001720194451		
innoproc_yes,newtothemarket	innoproc	-0.0001746030876		0
dissinf_yes,onanirregularbasis	dissinf	-0.0001754501532		0
overskilld_30.0	overskilld	-0.0001810735129		
mmerintrain_notatall	mmerintrain	-0.0001861150097	0	0
actprod_yes	actprod	-0.0001906829942	0	0
learnnoneedd_40%to59%	learnnoneedd	-0.0001910390835		0
tauton_teammembersdecideamongthemselves	tauton	-0.0001922323706	0	0
ictcompd_60%to79%	ictcompd	-0.0001952387388		0
pcwkmachd_20%to39%	pcwkmachd	-0.0002033665731		0
staffme_yes,onanirregularbasis	staffme	-0.0002034304403		0
smainactd_professional,scientificandtechnicalactivities	smainactd	-0.0002155033081		0
somedi_yes,onanirregularbasis	somedi	-0.0002292546681		0
comorgd_40%to59%	comorgd	-0.0002294500324		0
trinn_fairlyimportant	trinn	-0.000229641997	0	0
underskilld_30.0	underskilld	-0.0002311896408		
pmstartcust_3	pmstartcust	-0.0002313620199	0	0
chempfut_itwillstayaboutthesame	chempfut	-0.0002325795363	0	0
overskilld_5.0	overskilld	-0.0002378070966		
mmepinorg_nodecisionsweremadeinthisarea	mmepinorg	-0.0002386511415	0	0
mmepinorg_toasmallextent	mmepinorg	-0.0002389170571	0	0
compprobsd_20%to39%	compprobsd	-0.0002409409784		0
smainactd_financialandinsuranceactivities	smainactd	-0.0002420551333	0.0004430569238	-0.0002622127
wagessetboth	wagessetboth	-0.0002432481956	0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
ictcompd_80%to99%	ictcompd	-0.0002432493564		0
mmerindism_notatall	mmerindism	-0.0002437142342	0	0
supchek_managerscontrolwhetheremployeesfollowthetasksassignedtothem	supchek	-0.0002452012242	0	0
estsize_50to249employees	estsize	-0.0002462412064	0	0
overskilld_15.0	overskilld	-0.0002579106438		
underskilld_25.0	underskilld	-0.0002667618922		
skillsmatchd_65.0	skillsmatchd	-0.0002688882495		
contrd_noneatall	contrd	-0.0002691684312	0	0
contrd_60%to79%	contrd	-0.0002787130129		0
mmepintime_nodecisionsweremadeinthisarea	mmepintime	-0.0002800302907	0	0
overskilld_20.0	overskilld	-0.0002816161575		
innoprod_no	innoprod	-0.00028331291	0	0
wagessetexternal	wagessetexternal	-0.0002849538468	-1.34E-06	0
mmepinorg_toagreatextent	mmepinorg	-0.0002881013279	0	0
mmepintrain_toasmallextent	mmepintrain	-0.0002888541265	-6.24E-05	0
ictapp_no	ictapp	-0.0002907809574	1.15E-05	0
overskilld_22.0	overskilld	-0.0003012594516		
dissinf_yes,onaregularbasis	dissinf	-0.0003241215508		0
smainactd_wholesaleandretailtrade;repairofmotorvehiclesandmotorcycles	smainactd	-0.0003298261617		0
compprobsd_40%to59%	compprobsd	-0.0003467567392		0
ictrob_yes	ictrob	-0.0003610360369	-2.99E-05	0
overskilld_0.0	overskilld	-0.0004369230029		
mmerintrain_skipped	mmerintrain	-0.0004485359929		
eratt_skipped	eratt	-0.0005198175366		
mmerintime_notatall	mmerintime	-0.0005276386768	0	0
prodvol_ithasstayedaboutthesame	prodvol	-0.003241099243	-9.86E-05	0
mmepintime_nan	mmepintime		-9.43E-06	0
skillsmatchd	skillsmatchd		5.73E-05	0
mmepintrain_nan	mmepintrain		0	0
mmepindism_nan	mmepindism		0	0
mmepinpay_nan	mmepinpay		0	0
mmerintrain_nan	mmerintrain		0	0
mmerindism_nan	mmerindism		0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
mmerinorg_nan	mmerinorg		0	0
somedi_nan	somedi		0	0
dissinf_nan	dissinf		0	0
staffme_nan	staffme		0	0
regmee_nan	regmee		0	0
mmepinorg_nan	mmepinorg		0	0
eicomp_nan	eicomp		0	0
eidelay_nan	eidelay		0	0
actdede_nan	actdede		0	0
actprod_nan	actprod		0	0
chempfut_nan	chempfut		0	0
qwprel_nan	qwprel		0	0
ictcompd_nan	ictcompd		0	0
itperfmon_nan	itperfmon		0	0
itprodimp_nan	itprodimp		0	0
ictrob_nan	ictrob		0	0
ictapp_nan	ictapp		0	0
prodvol_nan	prodvol		0	0
mmerinpay_nan	mmerinpay		0	0
mmerintime_nan	mmerintime		0	0
lowmot_nan	lowmot		0	0
retainemp_nan	retainemp		0	0
sickleave_nan	sickleave		0	0
overskilld	overskilld		5.21E-05	0
eratt_nan	eratt		0	0
contrd_nan	contrd		0	0
skillch_nan	skillch		0	0
pcwkmachd_nan	pcwkmachd		0	0
paidtraind_nan	paidtraind		0	0
training_nan	training		0	0
learnnoneedd_nan	learnnoneedd		0	0
supchek_nan	supchek		0	0
tauton_nan	tauton		0	0
teasin_nan	teasin		0	0
underskilld	underskilld		-7.69E-05	0
comorgd_nan	comorgd		0	0
compprobsd_nan	compprobsd		0	0
pmstartcust_nan	pmstartcust		0	0

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
pmstratbq_nan	pmstratbq		0	0
pmstratlp_nan	pmstratlp		0	0
innoproc_nan	innoproc		0	0
indir_nan	indir		0	0
ertrus_nan	ertrus		0	0
pmstratnps_nan	pmstratnps		0	0
emporg_nan	emporg		0	0
trflex_nan	trflex		0	0
innomark_nan	innomark		0	0
trski_nan	trski		0	0
wpsupp_nan	wpsupp		0	0
onjobd_nan	onjobd		0	0
itperfmonuse_nan	itperfmonuse		0.0001622102588	0
trinn_nan	trinn		0	0
innoprod_nan	innoprod		0	0
trmot_nan	trmot		0	0
innoproc_yesnewtotheestablishmentbutnotnewtothemarket	innoproc		0.0003489067925	
paidtraind_40to59	paidtraind		0.0001211226386	
paidtraind_20to39	paidtraind		7.98E-05	
learnnoneedd_60to79	learnnoneedd		4.43E-05	
staffme_yesonanirregularbasis	staffme		7.36E-06	
regmee_yesonaregularbasis	regmee		0	
regmee_yesonanirregularbasis	regmee		0	
staffme_yesonaregularbasis	staffme		0	
dissinf_yesonaregularbasis	dissinf		0	
dissinf_yesonanirregularbasis	dissinf		0	
somedi_yesonaregularbasis	somedi		0	
somedi_yesonanirregularbasis	somedi		0	
ictcompd_20to39	ictcompd		0	
ictcompd_40to59	ictcompd		0	
ictcompd_60to79	ictcompd		0	
ictcompd_80to99	ictcompd		0	
ictcompd_lessthan20	ictcompd		0	
smainactd_artsentertainmentandrecreation	smainactd		0	

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
smainactd_electricitygassteamandairconditioningsupply	smainactd		0	
smainactd_professionalscientificandtechnicalactivities	smainactd		0	
smainactd_watersupplyseweragewastemanagementandremediationactivities	smainactd		0	
smainactd_wholesaleandretailtraderepairofmotorvehiclesandmotorcycles	smainactd		0	
learnnoneedd_20to39	learnnoneedd		0	
learnnoneedd_40to59	learnnoneedd		0	
learnnoneedd_80to99	learnnoneedd		0	
learnnoneedd_lessthan20	learnnoneedd		0	
paidtraind_60to79	paidtraind		0	
paidtraind_80to99	paidtraind		0	
onjobd_20to39	onjobd		0	
onjobd_40to59	onjobd		0	
onjobd_60to79	onjobd		0	
onjobd_80to99	onjobd		0	
contrd_80to99	contrd		0	
contrd_60to79	contrd		0	
compprobsd_20to39	compprobsd		0	
compprobsd_40to59	compprobsd		0	
compprobsd_80to99	compprobsd		0	
compprobsd_lessthan20	compprobsd		0	
comorgd_20to39	comorgd		0	
comorgd_60to79	comorgd		0	
comorgd_80to99	comorgd		0	
comorgd_lessthan20	comorgd		0	
pcwkmachd_20to39	pcwkmachd		0	
pcwkmachd_40to59	pcwkmachd		0	
pcwkmachd_60to79	pcwkmachd		0	
pcwkmachd_80to99	pcwkmachd		0	
pcwkmachd_lessthan20	pcwkmachd		0	
contrd_20to39	contrd		0	
contrd_40to59	contrd		0	
innoproc_yesnewtothemarket	innoproc		0	

Index	Feature	Mean_SHAP_Value_rf	Mean_SHAP_Value_lgb	Mean_SHAP_Value_xgb
innomark_yesnewtotheestablishmentbutnotnewtothemarket	innomark		0	
innoprod_yesnewtotheestablishmentbutnotnewtothemarket	innoprod		0	
innoprod_yesnewtothemarket	innoprod		0	
contrd_lessthan20	contrd		-9.12E-07	
comorgd_40to59	comorgd		-6.42E-06	
paidtraind_lessthan20	paidtraind		-1.56E-05	
innomark_yesnewtothemarket	innomark		-3.38E-05	
onjobd_lessthan20	onjobd		-4.10E-05	
compprobsd_60to79	compprobsd		-5.46E-05	

Python Code

```
"""thesis.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1r-67v-KjQVzqUB_wBzTWLF1aiVTqPKYL
"""

import pandas as pd
import numpy as np
from matplotlib import pyplot
import matplotlib.pyplot as plt
import seaborn as sns
import re
import os

from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import OneHotEncoder
from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor
from sklearn.preprocessing import LabelEncoder, MinMaxScaler

from xgboost import XGBClassifier, XGBRegressor
import lightgbm as lgb
from sklearn.metrics import accuracy_score
!pip install shap
import shap

path = '/content/drive/MyDrive/Uinversity
Files/Thesis/datasets/stata13/ecs2019_mm_ukds.dta'
df = pd.read_stata(path)

pd.set_option('display.max_columns', None)
df.head()

def stat(series : pd.Series):
  return series.value_counts(normalize = True, dropna = False)*100

df.shape

df[['profit', 'chemp']] = df[['profit', 'chemp']].astype(str)
updated_df = df[~(df['profit'] == 'Not applicable, our company is a not-for-profit organisation')
          & ~(df['profit'] == 'Skipped') & ~(df['chemp'] == 'Skipped')].copy()
```

```
updated_df = updated_df.replace({'Skipped': None, 'Out of range': None})

updated_df.shape

updated_df.loc[(updated_df['mmerconfirm_v4_9'] == 'Yes') | (
    updated_df['mmerconfirm_v3_9'] == 'Yes'), "body"] = 'body does not exist'
updated_df.loc[(updated_df['mmerconfirm_v4_9'] == 'No') | (
    updated_df['mmerconfirm_v3_9'] == 'No'), "body"] = 'body exists'

wages_set_external = ['canat', 'casec', 'careg']
wages_set_internal = ['cacom', 'caocc', 'caoth']
updated_df.loc[:, 'wagessetexternal'] = (updated_df[wages_set_external] == 'Yes').any(axis=1)
updated_df.loc[:, 'wagessetinternal'] = (updated_df[wages_set_internal] == 'Yes').any(axis=1)
updated_df.loc[:, 'wagessetboth'] = ((updated_df[wages_set_internal] == 'Yes').any(
    axis=1) & (updated_df[wages_set_external] == 'Yes').any(axis=1))

germanic_cluster = ['Austria', 'Germany', 'Netherlands']
updated_df['country'] = updated_df['country'].astype(str)
df_updated_germanic = updated_df[updated_df['country'].isin(germanic_cluster)]

df_updated_germanic.shape

np.round(df_updated_germanic['country'].value_counts(normalize = True).values,2)

df_updated_germanic['country'].value_counts()

digital = ['ictcompd','ictapp','ictrob','itprodimp','itperfmon','itperfmonuse']

collaboration = ['actprod', 'actdede']
df_updated_germanic.loc[:, collaboration] = df_updated_germanic[collaboration].applymap(
    lambda text: 'Yes' if isinstance(text, str) and 'Yes' in text else (
        text if not pd.isna(text) else text)
).copy()

# for column in ['skillsmatch_d', 'overskill_d', 'underskill_d']:
#     df_updated_germanic[column] = pd.to_numeric(df_updated_germanic[column],
errors='coerce')

df_updated_germanic[['skillsmatch_d', 'overskill_d', 'underskill_d']] = df_updated_germanic[[
    'skillsmatch_d', 'overskill_d', 'underskill_d']].apply(lambda col: pd.to_numeric(col,
errors='coerce'))

df_updated_germanic[['skillsmatch_d', 'overskill_d', 'underskill_d']].dtypes

df_updated_germanic.columns = [re.sub('_','',col) for col in df_updated_germanic.columns]
```

```
df_updated_germanic.columns

# Job complexity
# used without any changes
job_complexity = ['teamex', 'teasin', 'tauton',
               'supchek', 'compprobsd', 'comorgd', 'pcwkmachd']

# skill level
skills = ['skillsmatchd', 'overskilld', 'underskilld', 'skillch']

# training (without change)
training = ['contrd', 'learnnoneedd', 'training', 'paidtraind',
          'onjobd', 'wpsupp', 'trski', 'trflex', 'trinn', 'trmot']

# innovation(without change)
innovation = ['innoprod', 'innomark', 'innoproc']

#product_market_strategy(without change)
product_market_strategy = ['pmstratlp','pmstratbq','pmstartcust','pmstratnps']

#employee_voice(direct, indirect)
indirect_emp_part =
['emporg','body','wagessetinternal','wagessetexternal','wagessetboth','ertrus', 'indir', 'eratt']
direct_emp_part =
['regmee','staffme','dissinf','somedi','eidelay','eicomp','mmepinorg','mmepindism','mmepintrain'
,'mmepintime','mmepinpay','mmerinorg','mmerindism','mmerintrain','mmerintime','mmerinpay'
]

#firm charactristic_from_main_thesis
firm_char = ['prodvol','estsize','smainactd','sickleave','retainemp','lowmot','qwprel','chempfut']

collaboration = ['actprod','actdede']

all_lists =
[job_complexity,skills,training,innovation,product_market_strategy,indirect_emp_part,direct_
emp_part,firm_char,collaboration,digital]

# merging the lists
features = []

for lst in all_lists:
   features.extend(lst)

print(features)

# in total, 68 features are used.
```

```
len(features)

df_updated_germanic[features].select_dtypes(include = ['bool']).columns

y = df_updated_germanic['profit']
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)
y_encoded

# xgboost

# one hot encoding categorical and object features
X = df_updated_germanic[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
                                columns=onehot_encoder.get_feature_names_out(),
                                index=X.index)

X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)




y = df_updated_germanic['profit']
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)


model_xgbc_11 = XGBClassifier(num_class=5,  eta = 0.01, max_depth = 2,num_parallel_tree
= 5,random_state=2)

scores = cross_val_score(model_xgbc_11, X_transformed, y_encoded, cv=5,
scoring='accuracy')

print("Cross-validation scores:", scores)
print("Mean CV score:", scores.mean())

X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)
```

```
model_xgbc_11.fit(X_train, y_train)

y_pred_test = model_xgbc_11.predict(X_test)
accuracy_score(y_test, y_pred_test)

feature_importances = model_xgbc_11.feature_importances_

xgb_features_imp_raw = pd.DataFrame({
    'feature': X_transformed.columns.to_list(),
    'importance': list(feature_importances)
})

xgb_features_imp_raw.sort_values(by='importance', ascending=False, inplace=True)

# plt.figure(figsize=(10, 8))
# plt.barh(xgb_features_imp_raw['feature'], xgb_features_imp_raw['importance'])
# plt.xlabel('Importance')
# plt.title('Feature Importance')

try:
  os.mkdir('results')
  print('directory created')
except:
  print('directory already exists')

xgb_features_imp_raw.to_csv('results/xgb_features_imp_raw.csv', index = False)

xgb_features_imp_agg = xgb_features_imp_raw.groupby(
    xgb_features_imp_raw['feature'].apply(lambda x: x.split('_')[0])
).sum().sort_values(by='importance', ascending=False)

xgb_features_imp_agg.to_csv('results/xgb_features_imp_agg.csv')

"""# Light GBM"""

# one hot encoding categorical and object features
X = df_updated_germanic[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
                                columns=onehot_encoder.get_feature_names_out(),
                                index=X.index)
```

```
X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)
X_transformed = X_transformed.rename(columns = lambda x:re.sub('[^A-Za-z0-9_]+', '', x))



y = df_updated_germanic['profit']
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

model_lgbm = lgb.LGBMClassifier(num_leaves=31,
learning_rate=0.05,n_estimators=100,max_depth=2, random_state=2, verbose= -1 )

scores = cross_val_score(model_lgbm, X_transformed, y_encoded, cv=5, scoring='accuracy')
print("Cross-validation scores:", scores)
print("Mean CV score:", scores.mean())

X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)

model_lgbm.fit(X_train, y_train)

y_pred_test = model_lgbm.predict(X_test)
accuracy_score(y_test, y_pred_test)

feature_importances = model_lgbm.booster_.feature_importance(importance_type='gain')
feature_importances_normalized = feature_importances / sum(feature_importances)


features_df_lgbm_raw = pd.DataFrame({
    'feature': X_transformed.columns,
    'importance': feature_importances_normalized
})

features_df_lgbm_raw.sort_values('importance', ascending=False, inplace=True)
features_df_lgbm_raw.to_csv('results/features_df_lgbm_raw.csv', index = False)

features_df_lgbm_agg = features_df_lgbm_raw.groupby(
    features_df_lgbm_raw['feature'].apply(lambda x: x.split('_')[0])
).sum().sort_values(by='importance', ascending=False)

features_df_lgbm_agg.to_csv('results/features_df_lgbm_agg.csv')

print(pd.Series(y_pred_test).value_counts())
```

```
print(pd.Series(y_test).value_counts())

"""#random forest"""

# replacing all the null values with 'skipped' from the questionnaire

random_forest_df = df_updated_germanic[features].copy()
random_forest_df.loc[:,'profit'] = df_updated_germanic['profit']

non_numeric_columns = random_forest_df.select_dtypes(exclude = ['int','float']).columns

random_forest_df[non_numeric_columns] =
random_forest_df[non_numeric_columns].astype(str)

random_forest_df = random_forest_df.replace({'nan' : 'skipped'})
random_forest_df = random_forest_df.fillna('skipped')

# for col in random_forest_df.columns:
#   print(stat(random_forest_df[col]),'\n')

#feature transformation
X = random_forest_df[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

X[categorical_cols] = X[categorical_cols].astype(str)
onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
                              columns=onehot_encoder.get_feature_names_out(),
                              index=X.index)

X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)




y = random_forest_df['profit']
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

rf_classifier = RandomForestClassifier(random_state=2)
scores = cross_val_score(rf_classifier, X_transformed, y, cv=5, scoring='accuracy')
```

```
print("Cross-validation scores:", scores)
print("Mean CV score:", scores.mean())

X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)
rf_classifier.fit(X_train, y_train)

y_pred_test = rf_classifier.predict(X_test)
accuracy_score(y_test, y_pred_test)

feature_names = np.array(X_transformed.columns)
importances = rf_classifier.feature_importances_
type(importances)

rf_feature_imp_raw = pd.DataFrame({
    'feature': feature_names,
    'importance': importances
})
rf_feature_imp_raw = rf_feature_imp_raw.sort_values('importance', ascending = False)

rf_feature_imp_raw.to_csv('results/rf_feature_imp_raw.csv', index = False)

rf_feature_importances_agg = rf_feature_imp_raw.groupby(
    rf_feature_imp_raw['feature'].apply(lambda x: x.split('_')[0])
).sum().sort_values(by='importance', ascending=False)

rf_feature_importances_agg.to_csv('results/rf_feature_importances_agg.csv')

"""## Aggregating results

* pd concat all models aggregated
* creating the avg score of all
* finding the aggregating by list values
"""

results =
pd.concat([rf_feature_importances_agg,features_df_lgbm_agg,xgb_features_imp_agg], axis =
1, keys = ['random_forst','lgbm','xgboost'])
results.columns = ['{}_{}'.format(col[1], col[0]) for col in results.columns]
results.head()

results['avg_importance_score'] = results.filter(regex = 'importance').mean(axis = 1)

results.to_csv('results/final.csv')

#finding feature importance per category
```

```
feature_groups = {
    'job_complexity': job_complexity,
    'skills': skills,
    'training': training,
    'innovation': innovation,
    'product_market_strategy': product_market_strategy,
    'indirect_emp_part': indirect_emp_part,
    'direct_emp_part': direct_emp_part,
    'firm_char': firm_char,
    'collaboration': collaboration,
    'digitalization' : digital
}

group_score = {}
for group, features in feature_groups.items():
  for feature in features:
      score = float(results.loc[feature, 'avg_importance_score'])
      if group_score.get(group):
        group_score[group] += score
      else:
        group_score[group] = score

group_score = dict(sorted(group_score.items(), key = lambda x: x[1], reverse = True))
# feature_groups.items()

aggregated_group_results = pd.DataFrame(group_score.items(),
columns=['feature_group','aggregated_importance'])
aggregated_group_results

feature_names = [str(feature_groups[col]) for col in
aggregated_group_results['feature_group'].to_list()]

aggregated_group_results['feature_names'] = feature_names
aggregated_group_results

aggregated_group_results.to_csv('results/aggregated_groups.csv', index = False)

sns.barplot(aggregated_group_results, y = 'feature_group', x =  'aggregated_importance',
orient='h')
plt.show()

"""## Regressors

In this step, I convert the outcome variable, profit, from string to integer in order to find the
direction of each factor's effect on it.
"""
```

```
# merging the lists
features = []

for lst in all_lists:
    features.extend(lst)

print(features)

len(features)

"""**XGBoost**"""

# xgboost

# one hot encoding categorical and object features
X = df_updated_germanic[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
                                columns=onehot_encoder.get_feature_names_out(),
                                index=X.index)

X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)




y_encoded = df_updated_germanic['profit'].map({'No, we made loss' : -1, 'We broke even' : 0,
'Yes, we made a profit' : 1})


model_xgbr = XGBRegressor(eta = 0.01, max_depth = 2,num_parallel_tree = 5,
random_state=2)

#cross validation
scores = cross_val_score(model_xgbr, X_transformed, y_encoded, cv = 5 ,
scoring='neg_mean_squared_error')
print(-1*scores)
print(np.mean(-1*scores))
```

```
X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)
model_xgbr.fit(X_train, y_train)

y_pred_test = model_xgbr.predict(X_test)

feature_importances = model_xgbr.feature_importances_

xgb_features_imp_raw_reg = pd.DataFrame({
    'feature': X_transformed.columns.to_list(),
    'importance': list(feature_importances)
})

xgb_features_imp_raw_reg.sort_values(by='importance', ascending=False, inplace=True)

try:
  os.mkdir('results')
  print('directory created')
except:
  print('directory already exists')

xgb_features_imp_raw_reg.to_csv('results/xgb_features_imp_raw_reg.csv', index = False)

xgb_features_imp_agg_reg = xgb_features_imp_raw_reg.groupby(
    xgb_features_imp_raw_reg['feature'].apply(lambda x: x.split('_')[0])
).sum().sort_values(by='importance', ascending=False)

xgb_features_imp_agg_reg.to_csv('results/xgb_features_imp_agg_reg.csv')

# Calculate SHAP values for XGBoost
xgb_explainer = shap.Explainer(model_xgbr)
xgb_shap_values = xgb_explainer(X_train)
xgb_shap_df = pd.DataFrame(xgb_shap_values.values, columns=X_train.columns)
xgb_mean_shap_values = xgb_shap_df.mean()
shap.summary_plot(xgb_shap_values, X_train)

xgb_shap_df = pd.DataFrame(xgb_shap_values.values, columns=X_train.columns)
xgb_mean_shap_values = xgb_shap_df.mean()
xgb_mean_shap_values = pd.DataFrame(xgb_mean_shap_values,
columns=['Mean_SHAP_Value']).sort_values(by='Mean_SHAP_Value', ascending=False)
print(xgb_mean_shap_values)

xgb_mean_shap_values

xgb_mean_shap_values.to_csv('results/xgb_mean_shap_values.csv')
```

```
"""**LGBM**"""

X = df_updated_germanic[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
                              columns=onehot_encoder.get_feature_names_out(),
                              index=X.index)

X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)
X_transformed = X_transformed.rename(columns = lambda x:re.sub('[^A-Za-z0-9_]+', '', x))



y_encoded = df_updated_germanic['profit'].map({'No, we made loss' : -1, 'We broke even' : 0,
'Yes, we made a profit' : 1})

model_lgbm = lgb.LGBMRegressor(num_leaves=31,
learning_rate=0.05,n_estimators=100,max_depth=2, random_state=2, verbose= -1 )


scores = cross_val_score(model_lgbm, X_transformed, y_encoded, cv = 5 ,
scoring='neg_mean_squared_error')
print(-1*scores)
print(np.mean(-1*scores))

X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)

model_lgbm.fit(X_train, y_train)

feature_importances = model_lgbm.booster_.feature_importance(importance_type='gain')
feature_importances_normalized = feature_importances / sum(feature_importances)

features_df_lgbm_raw_reg = pd.DataFrame({
    'feature': X_transformed.columns,
    'importance': feature_importances_normalized
})
```

```
features_df_lgbm_raw_reg.sort_values('importance', ascending=False, inplace=True)
features_df_lgbm_raw_reg.to_csv('results/features_df_lgbm_raw_reg.csv', index = False)

features_df_lgbm_agg_reg = features_df_lgbm_raw_reg.groupby(
    features_df_lgbm_raw_reg['feature'].apply(lambda x: x.split('_')[0])
).sum().sort_values(by='importance', ascending=False)

features_df_lgbm_agg_reg.to_csv('results/features_df_lgbm_agg_reg.csv')

lgb_explainer = shap.Explainer(model_lgbm)
lgb_shap_values = lgb_explainer(X_train)
shap.summary_plot(lgb_shap_values, X_train)

lgb_shap_df = pd.DataFrame(lgb_shap_values.values, columns=X_train.columns)
lgb_mean_shap_values = lgb_shap_df.mean()
lgb_mean_shap_values = pd.DataFrame(lgb_mean_shap_values,
columns=['Mean_SHAP_Value']).sort_values(by='Mean_SHAP_Value', ascending=False)
print(lgb_mean_shap_values)

lgb_mean_shap_values.to_csv('results/lgb_mean_shap_values.csv')

"""**RandomForest**"""

random_forest_df = df_updated_germanic[features].copy()
random_forest_df.loc[:,'profit'] = df_updated_germanic['profit']

non_numeric_columns = random_forest_df.select_dtypes(exclude =
['int','float','bool']).columns

random_forest_df[non_numeric_columns] =
random_forest_df[non_numeric_columns].astype(str)


random_forest_df = random_forest_df.replace({'nan' : 'skipped'})
random_forest_df = random_forest_df.fillna('skipped')

#feature transformation
X = random_forest_df[features]
categorical_cols = X.select_dtypes(include=['category', 'object']).columns.tolist()

numeric_cols = X.select_dtypes(include=['float64', 'int','bool']).columns.tolist()

X[categorical_cols] = X[categorical_cols].astype(str)
onehot_encoder = OneHotEncoder(sparse_output=False)
X_categorical_transformed = onehot_encoder.fit_transform(X[categorical_cols])
X_categorical_df = pd.DataFrame(X_categorical_transformed,
```

```
                              columns=onehot_encoder.get_feature_names_out(),
                              index=X.index)

X_numeric_df = X[numeric_cols]
X_transformed = pd.merge(X_numeric_df, X_categorical_df, left_index=True,
right_index=True)

y_encoded = df_updated_germanic['profit'].map({'No, we made loss' : -1, 'We broke even' : 0,
'Yes, we made a profit' : 1})

rf_regressor = RandomForestRegressor(random_state=2)


scores = cross_val_score(rf_regressor, X_transformed, y_encoded, cv = 5 ,
scoring='neg_mean_squared_error')
print(-1*scores)
print(np.mean(-1*scores))

X_train, X_test, y_train, y_test = train_test_split(X_transformed, y_encoded, test_size=0.2,
random_state=2)
rf_regressor.fit(X_train, y_train)

y_pred_test = rf_regressor.predict(X_test)

feature_names = np.array(X_transformed.columns)
importances = rf_regressor.feature_importances_


rf_feature_imp_raw_reg = pd.DataFrame({
    'feature': feature_names,
    'importance': importances
})
rf_feature_imp_raw_reg = rf_feature_imp_raw_reg.sort_values('importance', ascending =
False)

rf_feature_imp_raw_reg.to_csv('results/rf_feature_imp_raw_reg.csv', index = False)

rf_feature_importances_agg_reg =
rf_feature_imp_raw_reg.groupby(rf_feature_imp_raw_reg['feature'].apply(lambda x:
x.split('_')[0]))\
.sum().sort_values(by='importance', ascending=False)

rf_feature_importances_agg_reg.to_csv('results/rf_feature_importances_agg_reg.csv')

explainer = shap.TreeExplainer(rf_regressor)
shap_values = explainer.shap_values(X_train)
```

```
shap.summary_plot(shap_values, X_train)

shap_df = pd.DataFrame(shap_values, columns=X_train.columns)
mean_shap_values = shap_df.mean()
rf_mean_shap_values = pd.DataFrame(mean_shap_values,
columns=['Mean_SHAP_Value']).sort_values(by='Mean_SHAP_Value', ascending=False)
print(rf_mean_shap_values)

rf_mean_shap_values.shape

rf_mean_shap_values.to_csv('results/rf_mean_shap_values.csv')

"""## Aggregating feature importances"""

results =
pd.concat([rf_feature_importances_agg_reg,features_df_lgbm_agg_reg,xgb_features_imp_agg
_reg], axis = 1, keys = ['random_forst','lgbm','xgboost'])
results.columns = ['{}_{}'.format(col[1], col[0]) for col in results.columns]
results.head()

results['avg_importance_score'] = results.filter(regex = 'importance').mean(axis = 1)

results.to_csv('results/final_regressors.csv')

#finding feature importance per category
feature_groups = {
    'job_complexity': job_complexity,
    'skills': skills,
    'training': training,
    'innovation': innovation,
    'product_market_strategy': product_market_strategy,
    'indirect_emp_part': indirect_emp_part,
    'direct_emp_part': direct_emp_part,
    'firm_char': firm_char,
    'collaboration': collaboration,
    'digitalization' : digital
}

group_score = {}
for group, features in feature_groups.items():
  for feature in features:
      score = float(results.loc[feature, 'avg_importance_score'])
      if group_score.get(group):
        group_score[group] += score
      else:
        group_score[group] = score
```

```
group_score = dict(sorted(group_score.items(), key = lambda x: x[1], reverse = True))
# feature_groups.items()

aggregated_group_results = pd.DataFrame(group_score.items(),
columns=['feature_group','aggregated_importance'])
aggregated_group_results

feature_names = [str(feature_groups[col]) for col in
aggregated_group_results['feature_group'].to_list()]

aggregated_group_results['feature_names'] = feature_names
aggregated_group_results

aggregated_group_results.to_csv('results/aggregated_groups_regressors.csv', index = False)

sns.barplot(aggregated_group_results, y = 'feature_group', x =  'aggregated_importance',
orient='h')
plt.show()

"""## Aggregating shap values"""

xgb_mean_shap_values.reset_index(inplace = True)

#xgb_mean_shap_values.drop(columns = ['level_0'], inplace = True)
xgb_mean_shap_values

lgb_mean_shap_values.reset_index(inplace = True)
lgb_mean_shap_values

rf_mean_shap_values.reset_index(inplace = True)
rf_mean_shap_values

rf_mean_shap_values['index'] = rf_mean_shap_values['index'].str.lower().str.replace(" ","")
lgb_mean_shap_values['index'] = lgb_mean_shap_values['index'].str.lower().str.replace(" ","")
xgb_mean_shap_values['index'] = xgb_mean_shap_values['index'].str.lower().str.replace("
","")

shap_raw_results = pd.concat([rf_mean_shap_values.set_index('index'),
xgb_mean_shap_values.set_index('index'),lgb_mean_shap_values.set_index('index')], axis = 1,
        keys = ['rf','xgb','lgb'])
shap_raw_results.columns = ['{}_{}'.format(col[1], col[0]) for col in
shap_raw_results.columns]

# shap_raw_results.head()
```

```
shap_raw_results = shap_raw_results.reset_index()
shap_raw_results.head()

shap_raw_results['feature'] = shap_raw_results['index'].apply(lambda x: x.split('_')[0])
shap_raw_results.head()

shap_raw_results.to_csv('results/shap_raw_results.csv', index = False)

shap_raw_results = shap_raw_results.drop(columns = 'index')
shap_raw_results.head()

shap_agg_results =
shap_raw_results.set_index('feature').abs().groupby('feature').sum().reset_index()

scaler = MinMaxScaler()

columns = ['Mean_SHAP_Value_rf','Mean_SHAP_Value_xgb','Mean_SHAP_Value_lgb']
shap_agg_results[columns] =
shap_agg_results[columns].div(shap_agg_results[columns].sum())

print(shap_agg_results[columns].sum())

shap_agg_results['avg_value'] = shap_agg_results.filter(regex = 'Value').mean(axis=1)
shap_agg_results.head(10)

shap_agg_results = shap_agg_results.sort_values('avg_value', ascending = False)
shap_agg_results.head()

shap_agg_results.tail(5)

shap_agg_results.to_csv('results/shap_agg_results.csv', index = False)

#finding feature importance per category
feature_groups = {
    'job_complexity': job_complexity,
    'skills': skills,
    'training': training,
    'innovation': innovation,
    'product_market_strategy': product_market_strategy,
    'indirect_emp_part': indirect_emp_part,
    'direct_emp_part': direct_emp_part,
    'firm_char': firm_char,
    'collaboration': collaboration,
    'digitalization' : digital
}
```

```
group_score = {}
for group, features in feature_groups.items():
  for feature in features:
      score = float(shap_agg_results[shap_agg_results['feature'] == feature]['avg_value'])
      if group_score.get(group):
        group_score[group] += score
      else:
        group_score[group] = score

group_score = dict(sorted(group_score.items(), key = lambda x: x[1], reverse = True))
# feature_groups.items()

aggregated_group_results_shap = pd.DataFrame(group_score.items(),
columns=['feature_group','aggregated_importance'])
aggregated_group_results_shap

aggregated_group_results_shap.to_csv('results/aggregated_group_results_shap.csv')
```