Working Paper.

UNITED NATIONS
ECONOMIC COMMISSION FOR EUROPE

CONFERENCE OF EUROPEAN STATISTICIANS

Workshop on Statistical Data Editing
(Neuchâtel, Switzerland, 18-20 September 2018)

Improving Data Validation using Machine Learning

Prepared by Dr. Christian Ruiz, Swiss Federal Statistical Office, Switzerland[1]

I. Preliminary Remarks

1. This working paper presents a very early idea that emerged mid-2017 when undertaking data quality controls within the statistics of staff on higher education institutions (HEI). Its responsibility is within the section for educational processes at the Swiss Federal Statistical Office (FSO). Basic tests showed quickly that there is a certain potential for further research. This year that endeavour became more formalized as one of five pilot projects of our statistical institutes' new 'Data Innovation Strategy'. However, the presented elements aim to communicate the underlying idea and are not formally part of the ongoing pilot project.

2. The new Data Innovation Strategy provides an opportunity for us to bring the idea into a more concrete realization and to think about how it could become part of the production process. It states: „The focus of the strategy is to augment and/or complement existing basic official statistical production at the Swiss Federal Statistical Office (FSO) in the areas where data innovation … makes sense." (FSO 2017: 5). It then defines "**data innovation** as the application of complementary analytics methods (e.g. predictive analytics using approaches from advanced statistics, data science and/or machine learning) to existing (or traditional) and/or new (or non-traditional) data sources" (FSO 2017: 9).

3. This complementary approach, we would like to illustrate using a slightly altered Venn diagram by Conway (see figure1)[2]. In that understanding 'Machine Learning' is the combination of, on the one hand, mathematical and statistical knowledge and, on the other hand, programming and IT skills. Experts that are rather specialized in one of these two fields might have different approaches, definitions, and understandings of the material at hand. If certain machine learning algorithms are used by IT experts as mere black boxes, there is a risk for grave mistakes. It is thus even more important to combine experts of both fields to enable a machine learning process that is both methodologically sound and based on elaborate IT solutions.

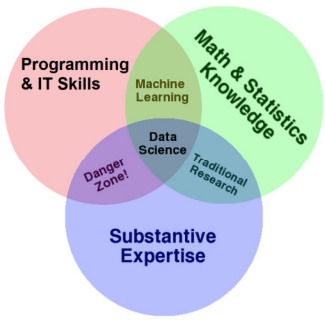

Figure1: Slightly adapted illustration originally by Drew Conway 2013

[1] I would also like to thank the core team of the pilot project and everybody that supported us.
[2] http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram (last accessed 19.06.2018)

4. In addition to these two circles, a third circle in the Venn diagram is required to obtain 'data science' - or in our case a final statistical and useful product. A machine learning application is eventually only as good, as it involves the knowledge of the expert that knows her/his data, its pitfalls, and its quality. She/he is after all the 'client' who knows best how such an application can serve her/him.

5. On the one hand, this complementary analytics approach explains why we will receive, as part of the pilot project, guidance by Prof. Diego Kuonen - for which we, as not trained methodologists, are very thankful as, even though we are a team of doctoral graduates and partly have a certified background in machine learning, we are not experts.

6. On the other hand, that approach also explains the aim of this working paper and participation in the workshop, which is to get a valuable feedback by an audience that consists of many trained statistical methodologists. We hope that this can give us a constructive input for the direction of the pilot project. We also hope to provide a certain humble inspiration for raising a certain interest in the matter.

II. Introduction

A. Data Validation

7. The pilot project performs machine learning in the area of data validation (DV)[3]. National statistical institutes (NSI) perform DV to test the reliability of delivered data. Data that seem either obviously wrong or possibly wrong is sent back to the data suppliers for correction or comment. This is particularly important in the domain of administrative data that is sent by data suppliers (such as communes, cantons/states, federal units, universities or schools), and where a two-way communication with these is possible and necessary as these data are not collected for statistical purposes in the first instant. Hitherto, such DV were performed in mainly two ways: either manually by eye-balling through the obtained data, or automatically through thresholds and logical tests. In certain cases the automatization (not performed by machine learning) is sufficiently advanced so that the data suppliers have to correct all the evident errors and have to give their OK for dubious cases. Therewith, the data that arrives at the NSI is of a higher quality, more quickly processed, and the process is more resource-efficient. We expect similar results with the application of machine learning in this domain – in a manner that might not replace but complement the existing automatic mechanisms.

8. The idea to perform machine learning on DV emerged rather by chance. Colleagues[4] from the same unit were discussing a certain surprising pattern in their data in a reunion with data suppliers. The surprising pattern turned out to be a marginal phenomenon that had good reasons to exist. This led to some reflexions about patterns in the data. Of the several ten thousand data points per year (and in other administrative data we are talking even of 6 or 7 digits per year) there might be 'right' patterns, there might be systematic mistakes, and there might be mistakes that are identifiable by humans to be wrong. The question thus emerged, whether such patterns and 'outliers' can be identified by a machine learning algorithm – or in other words by an automatic mechanism that goes beyond ex-ante defined thresholds or logical tests.

B. Machine Learning

9. As 'machine learning' is indeed currently a buzz word, we would like to define it in a useful way right from the start: machine learning, also known as "statistical learning" (Gareth et al. 2013), is a collection of "common and modern regression and classification techniques" for predictive analytics (Kuhn et al. 2013). It is a regression if quantitative data is to be predicted and a classification if categorical data is to be predicted. Further, machine learning is not entirely new: algorithms such as Fischer's linear discriminant analysis date back to 1936, other linear models emerged and became termed as generalized linear models in the 70s, and already in the 80s there were many non-linear algorithms such as classification and regression trees emerging, which was also linked to an increase of computation

[3] It is also known as plausibility checks.

[4] I warmly thank Elena Zafarana and Merlina Bajic for their support.

power (Gareth et al. 2013: 6). An additional increase of calculation power, of memory, of available data, and some important developments in research as well as open software permitted a very fertile soil in the last two decades for further progress in many areas – amongst them also in official statistics.

10. There are three different forms of machine learning algorithms. The one that will be used in this project and explained in the following paragraphs is based on **supervised learning**. The second form is unsupervised learning. **Unsupervised** algorithms are not used to predict data as they do not have an associated response variable. They are rather used for data analysis and to find patterns in the data. Examples of unsupervised learning algorithms are principal component analysis and cluster algorithms such as K-means. As will be later briefly mentioned, we are planning to implement a side-function for an unsupervised algorithm in order to check for anomalies that could occur in the production process. The third form is **reinforcement learning**. Recently this form of machine learning is gaining much popularity but it can only be used in few applications and not within our context.

11. A **supervised** machine learning algorithm is based on learning from past data in order to **predict - not explain -** new (yet-to-be-seen) data. This happens generally in two phases in case of a supervised learning algorithm. In the training phase, a substantially large amount of data is fed into an algorithm so that patterns can be learned to produce a predictive model. In a test phase, other data is used to apply the predictive model and check whether the accuracy of the prediction is high. The predictiveness of the model can be improved until a certain extent by continuously adjusting certain so-called hyper parameters.

12. There is a large number of different algorithms and they all work in very distinct ways. For regressions as well as classification a useful categorization is given in three subtypes: linear models, nonlinear models, and tree-based models (Kuhn et al. 2013). Some of these algorithms can be easily comprehended whilst other are more arcane. Textbooks with introductory purposes start often with well-known algorithms such as simple linear and logistic regressions to show basic underlying ideas (Gareth et al. 2013; Kuhn et al. 2013). After all, something that is common to all of the supervised algorithms is to have some predictive variables with which a dependent variable can be predicted. How this prediction is done, is the crux that distinguishes these algorithms. Analogies can often help to understand some of these algorithms: for example the task of drawing a simple frontier to separate the predicted classes (in the case of a support vector machine) or to draw trees or logical schemes (if age is below X, then Y).

13. In other, more advanced, algorithms such as random forests (Breiman 2001a) or stochastic gradient tree boosting (Friedman 2001) the algorithms are relatively complex. The general mechanism of the algorithms can be understood but the single steps between input and output are difficult to be traced in detail. This is why we underline that the aim of these algorithms is to **predict:** if an algorithm, even though it is highly complex and difficult to understand, performs well in its prediction then it fulfils its main objective.

14. These algorithms are thus not primarily used for generating **interpretability or explicability**, which can only be performed in other ways – mostly through the application of simpler algorithms. It is mathematically proven that it is impossible to combine high prediction performance and interpretability – something that is called 'Ockham's dilemma' or as Breiman argued: interpretability * accuracy < beta, where beta is a constant (see also Breiman 2001b, Shmueli 2010, and Shmueli and Koppius 2011). However, interpretability is nevertheless very important for our project. We will write below on how we intend to bridge the divide between prediction and interpretability.

III. The Application of the Machine Learning Algorithm

A. The Basic Mechanism

15. The main idea of this project is to use the already existing data in our system to train a supervised learning algorithm. A variable in the dataset is selected as dependent variable so that it is predicted by all the other independent variables in the dataset. As an example, we are using the data of staff working at higher education institutions (HEI). The dataset consists of over 70'000 data points per year and provides therewith an opportunity for an algorithm to learn from several hundred thousand data

points. The dependent variable chosen is the staff category, which can be one of four classes: professors, lecturers, research assistants, and administrative employees. The independent variables in our very first test set were sex, full-time equivalent (FTE), university field, age, nationality, and a binary variable to distinguish federal technical universities from cantonal universities.

$$\text{Staff category} = f(\text{sex, FTE, field, age, nationality, university})$$

16. After training the algorithm with the historic data, the new data entering the system is used as test data. The latter is done by inserting the independent variables of these data points into the algorithm to predict the chosen dependent variable. As a result, it is possible to compare the predicted values with the ones that were observed/delivered. As the algorithm does not only predict a value but also its corresponding posterior probability, it is now possible to find the data points that are, according to the algorithm, the most unlikely cases – or in other words the interesting cases that are the most likely to be *wrong* (an important caveat to the term 'wrong 'will follow later).

17. The following figure illustrates this with three hypothetical data points. On the left, the 6 independent variables are depicted. The next four columns correspond to the (posterior) probabilities calculated by the algorithm for the four staff categories. And on the right, the observed dependent variable is depicted. In the case of the first data point, the algorithm predicts with a probability of 89% that, given the data, the dependent variable is supposed to be a research assistant (label A), and the observed dependent variable is indeed a research assistant. The person is young (27 years old), works part-time, and in a technical field. It is very unlikely that that person is a professor or a lecturer. A research assistant position is indeed the category that seems the most fitting. In the second case, the predicted and the observed dependent variables match again. However, the probability is lower than before. And indeed, the person is one year younger and in the medical domain – which in Switzerland takes longer to finish. Even though the most likely is still a research assistant position, it becomes thus more likely that someone is having an administrative position. In the third case, the prediction seems to be wrong: the observed category is a professor, which was predicted with 35% as opposed to the most likely outcome with 52% as lecturer. And indeed, the person is older (57 years old), works part-time in a technical field. It seems that both would be plausible – either professor or a lecturer position.

Sex	FTE	Field	Age	Swiss	Uni	p(A|·)	p(D|·)	p(P|·)	p(U|·)	Observed
M	0.75	4.Exact	27	Yes	Yes	0.89	0.11	0.00	0.00	A ✓
F	0.80	5.Med.	26	No	Yes	0.66	0.34	0.00	0.00	A ✓
F	0.56	6.Techn.	57	No	No	0.06	0.07	0.35	0.52	P ⊘

Figure2: Three hypothetic data points showing the main idea of the mechanism

18. It is important to underline that even though a dependent variable is used, the message of a divergence between prediction and observed variable does not automatically imply that *that* particular variable is wrong. The message would rather be that *something* is wrong. If a 24 year old is coded as professor, it could be that that person is not a professor but it could also be that the age is wrong. The question of which variable is wrong comes in a second step further below under 'feedback-mechanism'.

19. What does *wrong* in this context mean? Of course this has to be treated with utmost care. Either the prediction could be wrong, the observation could be wrong, or the case is indeed a very unlikely case (for example a 26 year old professor). However, as we only will look at the most improbable cases, there should be a good reason why there is a divergence. Nevertheless, there will be a certain unavoidable amount of false positives.

20. Figure 3 shows the algorithm in practice with anonymized (slightly modified) data. It shows only the most probable mistakes. Through eye-balling we could indeed confirm that most of the top cases seem to be wrong. There are people in domains, where no academic staff should have been recorded, research assistants that are 18 year old, and people that are research assistants at the age of 72 and so on.

Sex	FTE	Field	Age	Swiss	Uni	Observed	Predicted	p(obs|·)	p(pred|·)
F	1.000	5. Medicine	18	TRUE	TRUE	A	D	0.002	0.996
M	1.000	5. Medicine	20	TRUE	TRUE	A	D	0.002	0.996
F	1.000	Other	18	TRUE	TRUE	A	D	0.007	0.989
F	1.000	Other	19	TRUE	TRUE	A	D	0.008	0.988
M	1.000	5. Medicine	21	TRUE	TRUE	A	D	0.009	0.988
F	0.200	8. Central	34	TRUE	FALSE	A	D	0.009	0.987
F	1.000	8. Central	22	FALSE	TRUE	A	D	0.01	0.987
F	0.900	8. Central	61	TRUE	TRUE	P	D	0.007	0.981
M	0.600	6. Technical	26	FALSE	FALSE	D	A	0.011	0.985
F	0.400	8. Central	31	FALSE	FALSE	A	D	0.011	0.984
F	1.000	5. Medicine	30	FALSE	TRUE	A	D	0.012	0.982
F	1.000	2. Economy	56	FALSE	TRUE	A	P	0.005	0.974
M	0.058	2. Economy	72	TRUE	TRUE	A	U	0.004	0.972
F	0.600	4. Exact	25	FALSE	FALSE	D	A	0.014	0.982
F	1.000	2. Economy	55	FALSE	TRUE	A	P	0.005	0.971
F	0.028	Other	55	TRUE	TRUE	D	U	0.008	0.973
M	0.700	4. Exact	25	FALSE	FALSE	U	A	0.002	0.967
F	0.021	Other	56	TRUE	TRUE	A	U	0.004	0.968
M	1.000	2. Economy	49	FALSE	TRUE	A	P	0.006	0.970
M	0.800	8. Central	36	FALSE	FALSE	A	D	0.016	0.980

Figure 3: Example data (slightly modified) of the most highly probable mistakes.

As can be seen, the (posterior) probabilities of the observed ('obs') labels are very small and, as such, something seems to be not right within the inputs.

21. In order to do such an analysis, there are certain requirements necessary. Five of them are of utmost importance: First, a large number of cases is required. This is necessary to train a well-performing machine learning algorithm. This is even more important if the number of variables is high. Second, there has to be a certain relation among the variables. If there are links between variables, an algorithm could learn to predict. For example, professors are rather of an older age, certain positions are stronger characterized by part-time employment, gender might have a link with certain domains, and so on. Third, these relationships should make sense in order to be applied. Imagine you would take sex as a dependent variable: An algorithm could predict with a high probability that a 70 year old professor in a technical domain working full-time is male. The issue here is that the training data does not contain many female professors of that age in technical domains, and would thus (technically correctly) reproduce this imbalance. Fourth, the feature engineering and pre-processing of the variables enable to obtain more useful information that a machine learning algorithm can subsequently process. In the previously shown example we used only some variables – but currently we are using feature engineering to prepare scores of variables that are supposed to help the algorithm to improve the generalization performance (i.e. predictive accuracy). This can also be done through the matching of datasets – for example by matching the staff of the higher education institutions with the datasets of students and university exams. If, say, someone recently graduated in linguistics, then she/he is much more likely to be either a research assistant or working for the administration than being already a professor or lecturer. Fifth, pre-processing is also important to modify the dataset into a format that is treatable by the algorithm (and it should be mentioned that there might be differences depending on the algorithms that are used).

22. Another important caveat concerns the fact that the training data unavoidably contains wrong data and that the algorithm could thus have certain biased effects by these partly tainted data. However, the above example shows that the algorithm was able to learn quite well and to predict with high accuracy the most likely mistakes – even though there must have been mistakes in the training process, too. The performance seems thus to be convincing for us – even though the caveat is quite substantial.

B. The Choice of the Machine Learning Algorithm

23. There are many supervised learning algorithms – which one is to choose? At a first testing phase, we tried to use as many classification techniques as possible, because the chosen dependent variable was a categorical variable (as opposed to regression techniques that have to be used with continuous variables). In doing so, we tried linear, non-linear, and tree-based algorithms (following the distinction in Kuhn et al. 2013). The main incentive was to choose the algorithm with the highest predictive power. The literature underlines that there is no best algorithm and that the performance very much depends on the

datasets. The following figure 4 shows the results of our early tests. Stochastic gradient tree boosting (in figure 4 denominated as T_GBM) predicted the data in over 92% correctly and with a narrow deviation.

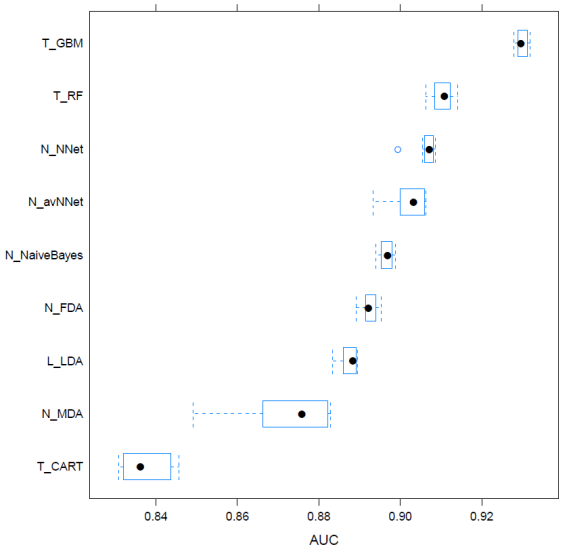

Figure 4: Results/Performance of different tested algorithms

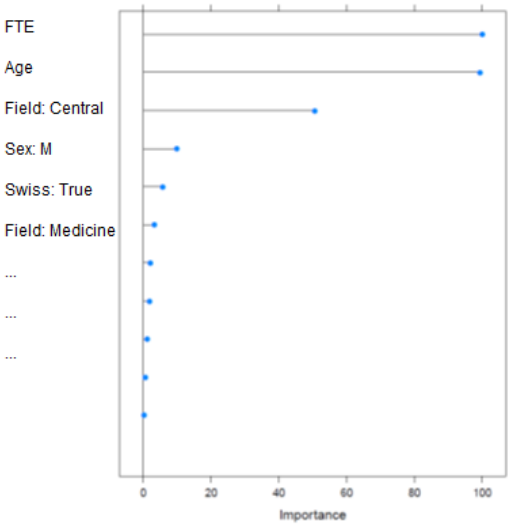

Figure 5: Variable importance with GBM

24. However, in the recent phase of the project we realized that there are more systematic questions that need to be fulfilled by the algorithm so that we can apply it. There are notably five such requirements. First, it is an algorithm that has to be able to process mixed-mode variables. Thus quantitative as well as categorical variables are used as inputs. Second, the algorithm should be able to calculate a variable importance. Such a variable importance is shown in figure 5 and ranks the variables according to their relative contribution to the predictive power. Third, the algorithm should handle outliers and in the ideal case should not be influenced by them. Fourth, it should have a low sensitivity to irrelevant input variables. And fifth, in the best case it should also be able to handle missing values. All of these requirements taken together, the group of tree-based algorithms fulfil them best.

IV. The Feedback-Mechanism

A. The Necessity of Interpretation and Explanation

25. Until now, the basic idea of a machine learning application for DV has been shown. This idea is based on prediction and indicates a probability that a data point might be wrong – or gives a hinge that something is odd. But in a production context, we cannot return data points to the data suppliers just arguing that it is very likely that these are wrong. For example, in the case of an automatic DV, the data suppliers are given information about the rules that are violated to certain data points. These rules might be 'hard' errors in case that, say, necessary information is lacking. Or they might be 'soft' warnings, when for example a threshold set for an age is transgressed – making it possible for the data supplier to override the warning and confirm that a certain data point is nevertheless correct.

26. Such a desired machine learning solution has to perform an operation similar to a manual 'feedback-mechanism'. However, this is precisely the issue that is connected to the above mentioned Ockham's dilemma – namely that an algorithm performs either very well in prediction or in interpretation/explanation. Therefore, an important part of this project is to find solutions to such a feedback-mechanism. While in a first step, the machine learning application explained above is meant to **predict** the most unlikely cases, a feed-back mechanism has in a second step to **explain** why these cases are likely to be wrong, and also what variables of these cases are most likely to be wrong. The following paragraphs present the current ideas that can be separated into two groups: those ideas that are based on a **global explanation** and those that are based on a **local explanation**. We are in midst of considering the feasibility as well as the pros and cons of the subsequently presented ideas, and do not favour yet any particular solution. We are very open to feedback from your side on these ideas.

B. First Group: Global Explanations

27. Global explanations are based on the entire dataset used to train the algorithm. The most often used tool for such a global explanation is the calculation of the **variable importances**, which shows the top-most important variables for the prediction - called 'the critical ones' in what follows. For example, figure 5 shows that FTE, age, and the central university field are the most important ones (from a predictive point of view), i.e. the critical ones, for the global explanation. Such a score of variable importance does not offer an explanation of the variables relevant in a specific case and thus it is not useful for a feedback-mechanism.

28. There is another idea connected to global explanations that could nevertheless be useful to convince the data-suppliers. It is based on **simple tree algorithms** or on algorithms that use rules using only the critical ones (see point above). As described above, there are advanced algorithms such as random forests or stochastic gradient tree boosting that perform well in prediction but that should not be used for interpretation. But using simple tree algorithms can serve to very easily describe/interpret the attribution to a certain class. The following figure shows a simple illustration of such a classification tree. The nodes (depicted as circles) stand for certain logical tests such as age>30 or sex=male and so on. The results of these tests are depicted in 'branches' going to a certain depth/height in order to obtain the distribution of the classes that fall in these branches/paths (depicted here with four bars, one for each staff category). The algorithm calculates these branches and logical tests by measures of purity (see for example Gareth et al. 2013). The desired depth/height of the algorithm can either be set automatically or manually. Rule-based algorithms work similar as tree-algorithms but are based on rules that are not dependant of a branch logic but rather on an intertwined if/else logic on many layers.

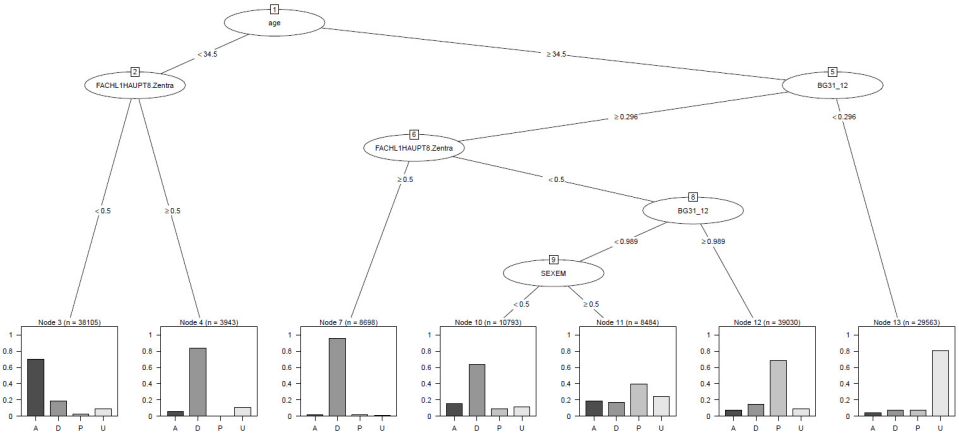

Figure 6: An example for a simple classification and regression tree

29. The added value of such an algorithm as a feedback-mechanism is its high capability to illustrate the paths leading to certain classes and its explanations, indicating why the chosen path is unlikely. It gives thus a certain tool-box for showing alternative paths for a case that is expected to be wrong. Such an explanation based on a logical path might be for some data suppliers more convincing than just an argumentation based on probabilities. However, it might also be difficult to be implemented when, say, several hundred cases are returned to a data supplier.

C. Second Group: Local Explanations

30. Local explanations, on the other hand, are more useful to explain specific cases – which is more appropriate to our objective to provide the data suppliers with a specific feedback for each case. The first idea in this group that we are trying, is based on the general idea of a **distance (or dissimilarity) matrix** (see for example Greenacre et al. 2014 chapters 5 and 6). A distance matrix provides the distance between data points. If, say, there are only two quantitative variables, then the distance can be easily understood as a simple Euclidean distance in a two-dimensional space. In a similar way, there are also ways to obtain distances between categorical variables – and even a collection of mixed-mode variables.

31. The application of such a distance matrix would be applied after the prediction mechanism explained above, which filters the most unlikely cases. But, looking at these cases, which are the variables that are most likely to be wrong? Using a distance matrix, the closest data point could be calculated in order to obtain the information about the variable that is different. Therewith, not only a probability can be sent back, but also the information about the most likely variable to be wrong and potentially even a value that might be suitable in that case.

32. An alternative, or rather a sub-form, to such a distance matrix is a **proximity matrix** such as it is produced by a random forests algorithm (Breiman 2001a). The added value of the proximity matrix is that it also uses the information of the variable importance to construct that special distance matrix. The relevant variable that could be wrong could thus be better identified. Further, the proximity matrix can be calculated as part of calculating the machine learning algorithm using random forests – and thus combining both in an elegant way. However, there is a technical large-memory issue connected to that due to its current implementation in R.

33. There is another more general caveat to the use of a distance matrix that currently is discussed within the project. What does it mean, if the closest data point is estimated? Is that data point a representative case for something like a cluster in that area? What if the closest case is just an outlier, or even worse, a similar mistake that is already contained in the data? Can such a distance matrix be improved by a density measure for cases with more similar neighbors?

34. A second idea, is based on certain R packages that recently emerged to tackle such issues linked to the explanation of complex algorithms such as **LIME**[5] and **DALEX**[6]. In the case of LIME it fits "a simple model around a single observation that will mimic how the global model behaves [in terms of 'goodness of fit, i.e. explanation, and not in terms of predictive accuracy] at that locality. The simple model can then be used to explain the predictions of the more complex model locally"[7]. Figure 7 shows an example of an output produced by LIME based on cancer prediction data[8]. In the top it shows that case 416 is predicted to be benign with a probability of 0.99 – so far this resembles the first part described above in our project. But then it shows the four most contributing variables in that very case. It is thus possible that we could use such a package to find the strongest contributing variable.

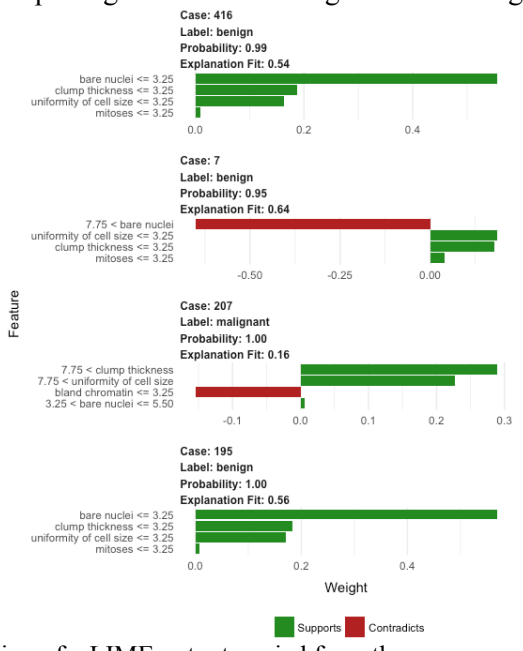

Figure 7: An illustration of a LIME output copied from the source mentioned in footnote 7

[5] https://cran.r-project.org/web/packages/lime/lime.pdf (last accessed on 4.7.2018)
[6] https://cran.r-project.org/web/packages/DALEX/DALEX.pdf (last accessed on 4.7.2018)
[7] https://cran.r-project.org/web/packages/lime/vignettes/Understanding_lime.html (last accessed on 3.7.2018)
[8] Ibid.

D. Further thoughts

35. Another idea is based on filtering the most unlikely cases given by the first predictive algorithm described in the last chapter, and then to apply for every variable another machine learning algorithm to obtain the probability for every variable in these filtered cases. In that way, the most problematic variable and its predicted value could be identified by the difference of predicted probabilities. We call this method **'brute force'** as the computing power to perform such a supervised machine learning algorithm on every variable would be considerable - or rather prohibitively large – notably when the number of variables is increased.

V. Further Methodological Issues

A. Balancing the Predicted Classes

36. One of the methodological issues that we encountered when assessing the usefulness of the above mentioned machine learning applications were unbalanced classes. Having unbalanced classes in the dependent variables has some important consequences for the prediction accuracy. The following figures illustrate this. Figure 8 depicts the so-called confusion matrix for the algorithm with unbalanced classes. A confusion matrix shows on the horizontal axis the observed values and on the vertical axis the predicted cases. The desired result would thus contain many cases in the diagonal – meaning that the cases were correctly predicted. Figure 9 shows the values of sensitivity and specificity for the four classes. As can be seen, the sensitivity for class professor ('P') is very low! Only in 46% of the cases where the observed variable was a professor, the prediction was a professor, too! This is a very low result and would pose a problem if not corrected.

	Reference			
Prediction	A	D	P	U
A	27013	3893	666	1402
D	2069	12255	976	1240
P	507	523	1781	417
U	1252	802	431	7190

Figure 8: Confusion matrix (ex-ante)

	Class: A	Class: D	Class: P	Class: U
Sensitivity	0.8759	0.7014	0.46212	0.7015
Specificity	0.8112	0.9047	0.97529	0.9524

Figure 9 : Sensitivity and specificity values (ex-ante)

37. However, the literature provides several ways to correct such an imbalance and to improve therewith the predictive power of the algorithm. If the algorithms allows, the methodologically best accepted way is to use priors in the sense of pre-defined (prior) weights. Current software packages that perform machine learning calculations allow easily performing such kind of priors' settings. Other two ways of correcting such imbalances, which can be prominently found in the machine learning literature, are downsampling and upsampling. In the case of downsampling, the amount of cases belonging to the same category in the training set is equalized. In the case of upsampling, the cases of certain underrepresented classes are used several times so to equalize the entire sample. We prioritize the use of priors, as this might be methodologically sounder. But downsampling and upsampling produced similar results.

38. The following two figures show an application of such a correction of imbalances. As can be seen in figure 11, the low sensitivity for class 'P' has been increased from 46% to over 81%!

	Reference			
Prediction	A	D	P	U
A	23427	2762	141	579
D	2837	11276	280	685
P	2455	2318	3130	1170
U	2122	1117	303	7815

Figure 10: Confusion matrix (ex-post)

	Class: A	Class: D	Class: P	Class: U
Sensitivity	0.7596	0.6453	0.81214	0.7625
Specificity	0.8897	0.9154	0.89852	0.9321

Figure 11 : Sensitivity and specificity values (ex-post)

B. Comparing to Baselines and the Production Environment

39. It became clear during the set-up of the pilot project that the machine learning DV is supposed to beat a certain baseline, in order to have a quantifiable measure that it performs better. Thus, an important part of the project is to prepare such baselines in order to compare the machine learning results against. We found out that this is not trivial. The question is notably: against what is it compared to? Until now

we did tests particularly connected to the ‘consolidated data’ that we have. But that means that the data were already filtered by existing automatic DV rules. Furthermore, there are other questions connected to the production environment that emerge.

40. One such question is where exactly the final machine learning DV should be built into the existing system. This is particularly the case as the solution will very likely be a combination of the current DV and the machine learning elements. For example, one could make the distinction between the ‘hard’ and ‘soft’ rules of the current DV. The hard cases have anyway to be filtered first –before it can be sent into a machine learning algorithm. So it might be that the baseline performance is just compared against the ‘soft’ part of the current DV.

41. Now, let us assume that the current DV provides 1’000 ‘soft’ warnings to the data suppliers that are considered to be wrong. Some of them are subsequently corrected (true positives as the algorithm predicted them correctly to be wrong) and some of them are overridden by the data supplier arguing therewith that the original provided data was correct (false positive). In the ideal case, the corrected cases can be calculated into the performance of the current DV. The overridden cases would in that ideal case be considered as false positives – which are by itself also a measure of performance of the system as it is one of the aims to keep the administrative work of our data suppliers low. However, this ideal case is unfortunately not representing the reality. We know that the data suppliers tend to be under time pressure and to try to finish the process of feedback loops as quickly as possible – which might result in an overriding of the warnings even though they might not be false positives. Indicators for that behaviour can for example be found when the time-stamps of the provided data are compared to the amount of data corrected. In some cases it is observable that data suppliers take more time to carefully go over the data and to correct some of the highlighted data points.

42. In a nutshell, these considerations belonging to an eventual final production environment have to be made already early on in the process. It is by all means very important that the situation of the final user is taken into account, which is the behaviour of the data supplier reacting, correcting or ignoring the feedback that she/he receives. Speaking metaphorically: without this consideration the final ‘client’ will not ‘buy the product’. The success of the project thus requires these considerations.

C. The Right Feature Engineering

43. Feature engineering and pre-processing are very important steps in the use of machine learning algorithms that should not be neglected as they are crucial for the performance and for a use that makes sense. As part of the pilot project we plan also to write a list of recommendations of best practices or general guidelines in how the features ought to be engineered. There might be three groups of considerations connected to feature engineering. The first one is linked to the use of the algorithms themselves that need the input variables to be in a certain format. Furthermore, there are differences among the algorithms. The second group of considerations is linked to software-considerations. Such considerations include the choice of the packages that partly perform already many machine learning tasks, or the types of variables chosen. The third group is linked to calculation power and memory issues. We indeed experienced first issues literally linked to large volumes of data and found some solutions to them. We are nevertheless still at a phase with many open questions and with many tests ahead of us.

D. Combination with an Unsupervised Learning Algorithm

44. Another consideration was the use of an unsupervised learning algorithm. Such an algorithm might help to identify expected patterns, systematic mistakes (unexpected patterns) or to observe cases that do not fall into any patterns. However, we currently rather think that an unsupervised learning algorithm could be applied in a different way than the basic mechanism described above. In our suggestion, an unsupervised learning algorithm could be used to periodically observe the data (which can also be during the production process) in order to observe new systematic errors or interesting outliers. An added value might also be the idea that a distance matrix or proximity matrix can serve for producing the data that can then subsequently be prepared instantaneously for analysis without further ado.

VI. Outlook

A. Issues to solve/discuss

45. As presented in this working paper, the ideas are still in a very early phase. There are thus many methodological questions that are still open and where we are currently trying to find practicable ways to set-up such a machine learning algorithm. We would be very grateful to receive feedback from your side on the basic mechanisms, on the proposed solutions for the feedback-mechanism (for example: does the idea with a distance matrix convince?), and also on any of the more methodological elements.

B. Next Steps

46. The pilot project now became formalized and it is expected to deliver a report of feasibility already by March 2019. By that time we need to answer many of those open questions, and advance our tests connected to the 'consolidated data'. Only after that, we will be able – if at all hopefully – to move further to tests linked to a production environment.

47. If machine learning should become part of a DV with the plausibilization of survey or administration data, a communication concept is needed. Such a concept, made in order to convince stakeholders on the usefulness of machine learning procedures need to be tailored according to the methodological knowledge of the different groups.

VII. Bibliography

-Breiman, Leo (2001a), Random Forests, *Machine Learning*, Vol. 45(1): 5—32.
-Breiman, Leo (2001b), Statistical Modeling: The Two Cultures, *Statistical Science*, Vol. 16(3): 199—231.
-Friedman, Jerome H. (2001), Greedy function approximation: A gradient boosting machine, *Annals of Statistics*, Vol. 28(5):1189—1232.
-FSO (2017), *Swiss Federal Statistical Office Data Innovation Strategy*, Federal Statistical Office: Switzerland.
https://www.bfs.admin.ch/bfs/en/home/news/whats-new.assetdetail.3862240.html
-Gareth, James & Witten, Daniela & Hastie, Trevor & Tibshirani, Robert (2013), *An Introduction to Statistical Learning*, 8th corrected edition 2017, Springer: New York.
- Greenacre, Michael & Primicerio, Raul (2014), *Multivariate Analysis of Ecological Data*, Fundacion BBVA: Spain.
-Kuhn, Max & Johnson, Kjell (2013), *Applied Predictive Modeling*, Springer: New York.
-Shmueli, Galit (2010), To Explain or to Predict?, *Statistical Science*, Vol. 25(3): 289—310.
-Shmueli, Galit & Koppius, Otto R. (2011), Predictive Analytics in Information Systems Research, *MIS Quarterly*, Vol. 25(3), 553—572.