Course project report

Feature Significance Analysis of the US Adult Income Dataset

AUTHORS: Junda Chen

Abstract

In this paper, we analyze the classic US Adult Income Dataset [1] using logistics regression and random forest to analyze potential factors that contribute to income bias for the $50K$ income bracket (income $\geq 50K$ per year). Using the two methods, we train the dataset and obtain stable models over cross validation. We also found that the two methods, although both showing good accuracy, exhibit conflicting interpretation about what factors have the most influence on the US adult income.

Code available at https://github.com/GindaChen/cs760-fa20

1 Introduction

US Adult Income Dataset (1994) [1] is a classic dataset that has been widely used in tutorial-based machine learning tasks such as Kaggle [1]. The classical approach is to use the dataset to train a model that predicts the income bracket with other information about the individual, without knowing how the model interprets the significance of each factor. The absence of "why" can trick people to prefer higher accuracy model without any specific reason to compare with other models that they discarded.

In this report, we seek to understand the feature significance of the US adult income dataset by analyzing the feature significance using logistics regression and random forest, the naive models that machine learning green hands tend to use while achieving similarly good accuracy on the dataset. During the process, we find the two models return drastically different interpretation on the how different features influence income dataset.

This report is organized as follows. Section 2 introduces the related works on the dataset. Section 3 explains the methodology of pre-processing and training. Section 4 shows the result of the two model. Section 5 discuss our observations and future work.

2 Related Works

Kaggle is probably one of the most representative hub that hosts the best answers of this challenge. Among them, we see methods range from logistic regression to boosting methods like Ada Boost and XGBoost, both used in a wide range of scientific works. To the best of our knowledge, most of the work in Kaggle focuses on the classical 50K-income-ladder prediction problem, and none of the work focus on the analysis of systematic bias of income. Worth mentioning, a plethora of works in other fields of science adopt the similar tool-chains in data analysis [1–37]. We learn from these works and adopt some of the methodologies in this project.

There are a good amount of work in the field of macro-economic that studies how different factors can

[1] US Adult Income Kaggle Challenge Page: https://www.kaggle.com/johnolafenwa/us-census-datax

influence the income distribution, using the full dataset provided in the US Census Bureau [2]. For simplicity, we only consider how the machine learning model tells us about the relationship between features and label.

3 Dataset

3.1 Dataset Overview

This dataset is a sample from the **US Census Database** that contains the census result of year 1994. It contains 48842 (not `NaN`) entries, and each entry contains the following features about a representative individual in the census record:

- **age**: (continuous, positive integer) The age of the individual.
- **workclass**: (categorical, 9 distinct values) Simplified employment status of an individual
- **fnlwgt**: (continuous, positive integer) Final weight of the record. Basically interpret as the number of people represented by this row.
- **education-num**: (categorical, 13 distinct values) The education level, in ascending positive integer value.
- **education**: (categorical, 13 distinct values) The education level. *Note that for simplicity, we will ignore this column because of the existence of education-num column.*
- **marital-status**: (categorical, 7 distinct values) Marital status of a person.
- **occupatioin**: (categorical, 15 distinct values) Rough category of the occupation.
- **relationship**: (categorical, 6 distinct values) Relationship in terms of the family. Note that we ignore this column since the semantic is somewhat covered by marital-status and gender.
- **race**: (categorical, 5 distinct values) Race of the person.
- **gender**: (boolean) gender at-birth.
- **capital-gain**: (continuous) Dollar gain of capital.
- **capital-loss**: (continuous) Dollar loss of capital.
- **hours-per-week**: (continous positive integer) Working hours per week.
- **native-country**: (categorical, 41 distinct values) Country at birth.
- **income-bracket**: (boolean) True if $\geq 50K$, otherwise False ($< 50K$ per year).

For simplicity of presentation, we ignore the exact distinct values in the paper. Reader can refer to the notebook for more details of the actual distinct values of each columns.

[2] US Census Bureau: https://www.census.gov/en.html

3.2 Nomenclatures

According to the context, we use the following symbols and terms:

- N: Sample size (i.e. number of rows in the dataset).
- X: The feature matrix after one-hot-encoding.
- y: The label column. $dim(y) = N \times 1$.
- w: The weight vector. Each row of the dataset is associated with a weight that specifies the number of people this sample can represent.
- *population*: Refer to the weight-adjusted number. For example, there are 2 records where $Age = 20$, each with weight 20 and 40. Then the population whose $Age = 20$ becomes $20 + 40 = 60$.

4 Methodology

4.1 Data Preprocessing and Toolchain

Feature Selection. We review the columns and discarded or simplified some columns.

- **Use `education-num` instead of `education`.** In between the `education` and `education-num`, we reserve `education-num` as it is an the integer representation marks the relative importance of education. Although this importance relation might not be linear, we still think it is better to treat the feature as non-categorical.
- **Binarize `naitive_country`**. We also binarize the `naitive_country` column such that only the value `United-States` is evaluated to `True`. This our intuition that `race` might be more representative features than the 40+ categorical values in countries. We kept the feature as race and native country are not necessarily strongly-related.
- **Filter features with small sample size**. One of the challenge we found during analysis is that some features that has ambiguous causal relation with the label tend to dominate the analysis because of the size of its sample. For example, the category `Occupation - Others Services` has only a few samples, hence it naturally tends to become a very discriminating feature. To avoid this phenomenon, we hand-filtered some categorical features that has very small sample sizes. This helps us interpreting the result, with sacrifice of the number of features we put into training.

One-hot encoding. We performed one-hot encoding on the rest of the categorical features (marital_status, occupation, race, workclass), which expand the feature matrix X to around 40 columns. We will use this matrix in our analysis.

Environment. We conduct all the computation on a Jupyter Notebook instance hosting on a MacBookPro with `Python 3.8`. We leverage machine learning using the open-sourced `scikit-learn` package (version '0.23.2') **cite here**. We also used `pandas` for data preprocessing, and `matplotlib` for data visualization.

4.2 Method 1: Logistics Regression

We use `LogisticRegression` to conduct logistics regression. We tune the following hyper-parameters to make logistic regression work as expected:

- Distance function: L2-norm
- tol: 10^{-3} to 10^{-7}. Empirical observation shows that accuracy tends to plateau when we reach 10^{-5}. Thus we eventually choose 10^{-5} for time-accuracy trade-off.
- max iteration: 10000 as empirically observed.
- regularization strength: 0.001 to 10. We explored the space and eventually set to the default value 1.

We perform cross validation using the function provided by scikit-learn. To analyze the significance of a feature, we inspect the coefficient (`coef_`) of each feature for each model, and analyze it accordingly. **Result is shown in the Section 5.1**

4.3 Method 2: Random Forest

We use `RandomForestClassifier` to train random forests. We tune the following hyper-parameters:

- criterion: We used gini impurity as the split criteria.
- max depth: We tried to vary the maximum depth from 4 to 8, but were not able to comprehend the benefit to fix a certain depth for the model. We finally decided to let the classifier choose the depth automatically.

We also perform cross validation on random forest. To analyze the significance of a feature, we inspect feature importance (`feature_importance`) of the forest for each forest we generate, and analyze the ranking of the features accordingly. **Result is shown in the Section 5.2**

4.4 Other Methods We Tried

LASSO. Lasso performed poorly on the dataset. Specifically, the evaluation on the one-hot encoded training set only yield an accuracy around 35%. We conjecture two reasons:

- (1) Features are not linearly related to the label, and as a result Lasso cannot determined the significance of one feature with respect to the label. This reason seems intuitive at first, but it does not explains why logistics regression performs well enough.
- (2) The feature space is not large enough for selection. Lasso performs well at gene selection where 1000+ weak genes are presented and only a few (4-10) genes are significantly related to the behaviour. We cannot know whether this is the reason as we are unable to extend the dataset large enough for analysis.

Baysian Network. We tried to use Baysian network to analyze the causality among different features. We use the naive Baysian classifier (`CategoricalNB`) in the scikit learn package to perform causality analysis, and have a pretty good preliminary training accuracy (about 80%). Sadly, we do not know how to visualize the Baysian network or analyze the causality relation between variables. In this report, therefore, we decide to not show case our result using Baysian network.

Neural Network (Multi-Layer Perceptron). We setup a series of multi-layer perceptrons with fully-connected hidden layers. Due to our naiveness of hyper-parameter tuning, we did not get a good result on the dataset (average accuracy 40%, very bad for a binary classification problem).

5 Result

5.1 Logistics Regression

Figure 1 shows the significance of the features in logistics regression. The absolute value of the coefficient in Figure 1(a) shows the significance of that feature, while its sign represents whether the feature is positively related to the label or not. 1(b) visualize the significance in ranking order.

Some of the features at the top ranking make sense for us to interpret. For example, `workclass without pay` (rank 1) definitely has a very significant negative relation to the individual's earning. In addition, Estimating the massive percentage of `Black` (rank 3) community also convince us that this feature is having a reasonable ranking.

Some features are top because of their category is under-representative. `race` such as American Indian or Eskimo (rank 2) is under-representative, and has a very low sample rate in the distribution. Some categorical feature with the value `Other` (rank 4, 5) also rank at top because they only represent a fraction amount of people.

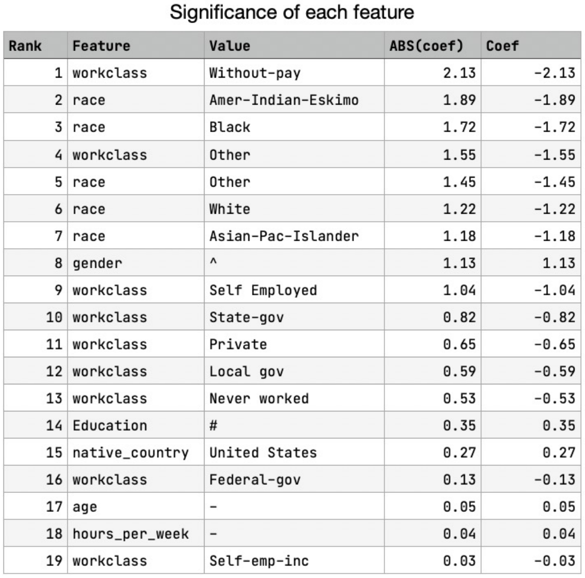
The logistics regression has its limitation when it comes to non-binary impact of a feature. For example, the impact of `Education` (rank 14) is well under-estimated in the model as we expected. We do not know the exact reason, but conjecture it is the problem of the dataset distribution.

Figure 2 shows the scores of cross validation at different iterations, and also the convergence process. As shown in the graph, the graph is near converged at 800 iterations, although at around 100 iterations it has showed the sign of convergence. We cross validate the model from the model of 100th iteartion to 100,000th iteration, and the result shows not much difference at the cross validation score (around 80%). We don't know if this shows the stability of the model or the naiveness of our problem using logistics regression, but we hope we can naively conclude that using logistics regression gives us good performance at the prediction task.

5.2 Random Forest

Figure 3(a)(b) shows the feature significance obtained from the random forest classifier. At the top, we see features like age, education, hours per week occupying the first 80% of the importance echelon. The distribution empirically looks like Zipf's law, although we did not verify the similarity. Figure 4 visualize one representative of the decision tree. Figure 3(c) shows that decision tree has a stable accuracy around 80%, close to the result of logistics regression (80%). The result does not change much even if we alter the number of estimators and maximum depth of the tree.

Our random forest model has a preference over integer values such as age, education number and hours per week. As a result, it shows completely opposite prediction to the logistics regression. We are surprised and puzzled that it does not pick extremely discriminating features (such as samples with `workclass` in the `Others` category) but instead features with a relatively even distribution (`age`, `education_num`).



Significance of each feature

Rank	Feature	Value	ABS(coef)	Coef
1	workclass	Without-pay	2.13	-2.13
2	race	Amer-Indian-Eskimo	1.89	-1.89
3	race	Black	1.72	-1.72
4	workclass	Other	1.55	-1.55
5	race	Other	1.45	-1.45
6	race	White	1.22	-1.22
7	race	Asian-Pac-Islander	1.18	-1.18
8	gender	^	1.13	1.13
9	workclass	Self Employed	1.04	-1.04
10	workclass	State-gov	0.82	-0.82
11	workclass	Private	0.65	-0.65
12	workclass	Local gov	0.59	-0.59
13	workclass	Never worked	0.53	-0.53
14	Education	#	0.35	0.35
15	native_country	United States	0.27	0.27
16	workclass	Federal-gov	0.13	-0.13
17	age	-	0.05	0.05
18	hours_per_week	-	0.04	0.04
19	workclass	Self-emp-inc	0.03	-0.03

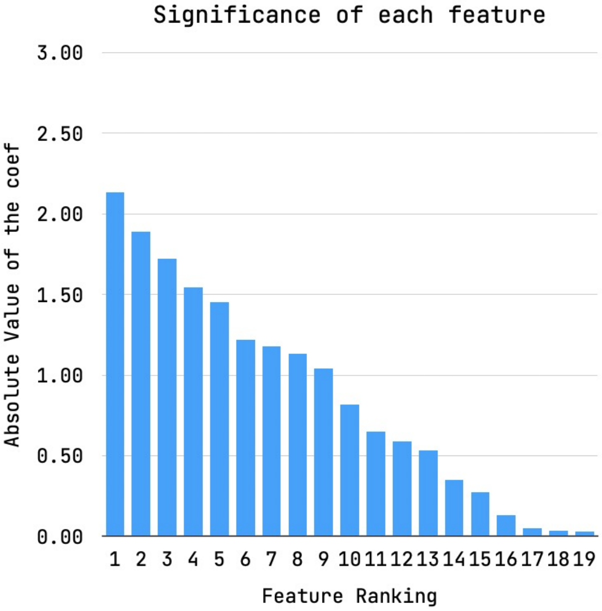
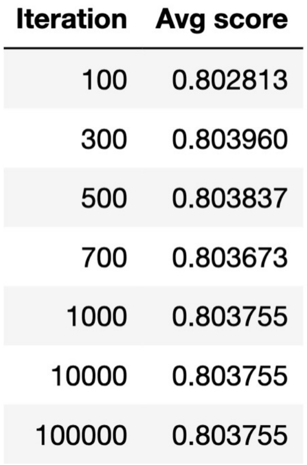


Figure 1: Feature significance using logistics regression. (a) Sort the coefficient (`coef`) by its absolute value (represented in `abs(coef)`). If the feature is categorical, a value is specified as its related one-hot encoded column feature; otherwise, the value is continuous if marked `-`, and integer if marked `#` . (b) Visualize the significance of each feature in ranking order.



Iteration	Avg score
100	0.802813
300	0.803960
500	0.803837
700	0.803673
1000	0.803755
10000	0.803755
100000	0.803755

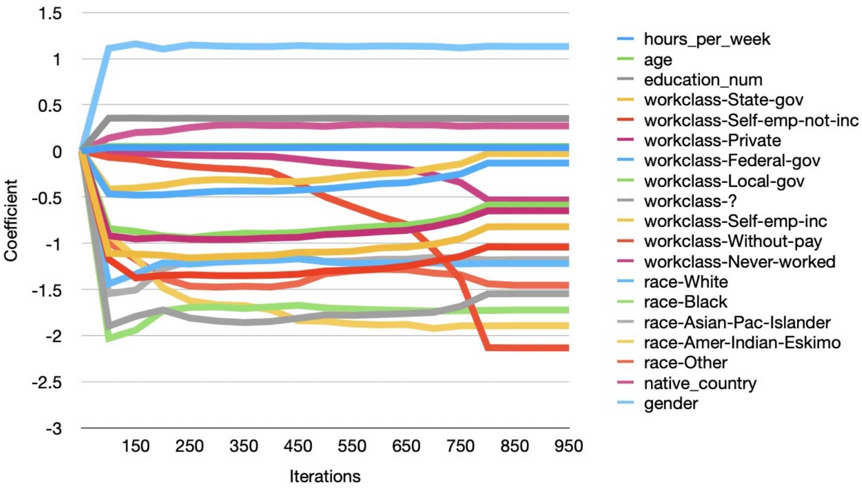
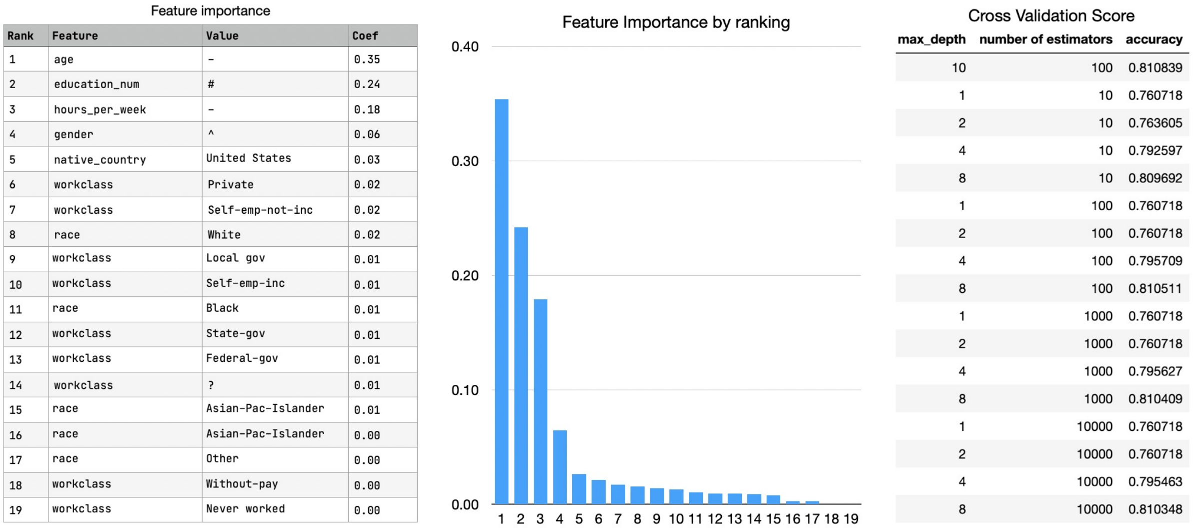


Figure 2: Cross validation score and the Convergence process of the logistic regression. (a) The average score of cross validation does not change significantly as more iterations goes on. (b) The graph zooms in the value of coefficient for the first 1000 iterations, plotting a value for each 50 steps.



Feature importance

Rank	Feature	Value	Coef
1	age	-	0.35
2	education_num	#	0.24
3	hours_per_week	-	0.18
4	gender	^	0.06
5	native_country	United States	0.03
6	workclass	Private	0.02
7	workclass	Self-emp-not-inc	0.02
8	race	White	0.02
9	workclass	Local gov	0.01
10	workclass	Self-emp-inc	0.01
11	race	Black	0.01
12	workclass	State-gov	0.01
13	workclass	Federal-gov	0.01
14	workclass	?	0.01
15	race	Asian-Pac-Islander	0.01
16	race	Asian-Pac-Islander	0.00
17	race	Other	0.00
18	workclass	Without-pay	0.00
19	workclass	Never worked	0.00

Feature Importance by ranking

Cross Validation Score

max_depth	number of estimators	accuracy
10	100	0.810839
1	10	0.760718
2	10	0.763605
4	10	0.792597
8	10	0.809692
1	100	0.760718
2	100	0.760718
4	100	0.795709
8	100	0.810511
1	1000	0.760718
2	1000	0.760718
4	1000	0.795627
8	1000	0.810409
1	10000	0.760718
2	10000	0.760718
4	10000	0.795463
8	10000	0.810348

Figure 3: Feature significance using the random forest. (a) Rank the feature significance by its `ranking`. (b) Visualize the significance of each feature by its ranking. (c) The cross validation accuracy with different hyper parameters.

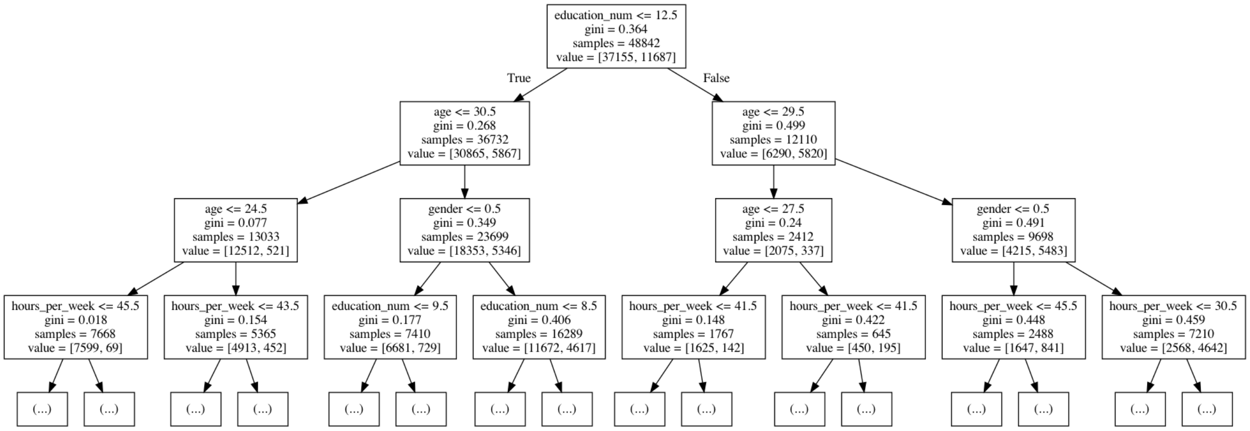


Figure 4: One representative tree in the random fores, showing the maximum depth = 3. This tree is consistent with the feature importance ranking: age, education, hours per week, and gender make the most impact in the decision process.

6 Discussion and Future Work

In Section 4 and 5, we show the feature significance using logistics regression model and random forest. These two models give completely opposite result on the dataset, while achieving similar accuracy through cross validation. We discuss the reason here and some future work to verify this phenomenon.

Pre-processing and model behaviour. Unlike the notebooks in Kaggle, we performed a one-hot encoding for all categorical features. This makes some column naturally discriminating. For logistic regression, we think this would help the model select categorical features independent of the other values in the same category; for random forest, we were afraid it would pick the discriminating columns at the very top. In reality, it turns out to be the opposite: regression model picks those that discriminates the most, and random forest picks the columns with more distinct values. So far, we have not understand why these models behave in the way it is.

Feature engineering. We also filtered out some of the features that have strange causal relation with the labeling feature. For some features, we also perform a binarization (e.g. `native_country`) such that the number of classes does not explode.

We think the way we engineer features may have affected the outcome of the model. We tried to rearrange the features or subdevide the feature categorization, but did not obtain a drastically different result. Other categorizations does not obey the causal relation, so we did not perform experiments on them.

Future Works. We aim to create a pipeline that can thoroughly analyze how the aforementioned factors can influence the dataset and different model, and how the interpretation will change with respect to these factors.

References

[1] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository.* 2017. URL: `http://archive.ics.uci.edu/ml`.

[2] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning.* MIT press, 2018.

[3] Bozhao Qi, Peng Liu, Tao Ji, Wei Zhao, and Suman Banerjee. "DrivAid: Augmenting driving analytics with multi-modal information". In: *2018 IEEE Vehicular Networking Conference (VNC).* IEEE. 2018, pp. 1–8.

[4] Wei Zhao, Liangjie Xu, Zhijie Sasha Dong, Bozhao Qi, Lingqiao Qin, et al. "Improving transfer feasibility for older travelers inside high-speed train station". In: *Transportation Research Part A: Policy and Practice* 113 (2018), pp. 302–317.

[5] Peter Harrington. *Machine learning in action.* Simon and Schuster, 2012.

[6] Jude W Shavlik, Thomas Dietterich, and Thomas Glen Dietterich. *Readings in machine learning.* Morgan Kaufmann, 1990.

[7] Wei Zhao, Tianxin Li, Bozhao Qi, Qifan Nie, and Troy Runge. "Terrain Analytics for Precision Agriculture with Automated Vehicle Sensors and Data Fusion". In: *Sustainability* 13.5 (2021), p. 2905.

[8] Lei Kang, Wei Zhao, Bozhao Qi, and Suman Banerjee. "Augmenting self-driving with remote control: Challenges and directions". In: *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications.* 2018, pp. 19–24.

[9] Bozhao Qi, Wei Zhao, Haiping Zhang, Zhihong Jin, Xiaohan Wang, and Troy Runge. "Automated traffic volume analytics at road intersections using computer vision techniques". In: *2019 5th International Conference on Transportation Information and Safety (ICTIS).* IEEE. 2019, pp. 161–169.

[10] Pat Langley. *Elements of machine learning.* Morgan Kaufmann, 1996.

[11] Issam El Naqa and Martin J Murphy. “What is machine learning?” In: *machine learning in radiation oncology.* Springer, 2015, pp. 3–11.

[12] Wei Zhao, Liangjie Xu, Jing Bai, Menglu Ji, and Troy Runge. “Sensor-based risk perception ability network design for drivers in snow and ice environmental freeway: a deep learning and rough sets approach”. In: *Soft computing* 22.5 (2018), pp. 1457–1466.

[13] Kevin P Murphy. *Machine learning: a probabilistic perspective.* MIT press, 2012.

[14] Yanping Hao, Liangjie Xu, Bozhao Qi, Teng Wang, and Wei Zhao. “A machine learning approach for highway intersection risk caused by harmful lane-changing behaviors”. In: *CICTP 2019.* 2019, pp. 5623–5635.

[15] Wei Zhao, Liangjie Xu, Shaoxin Xi, Jizhou Wang, and Troy Runge. “A sensor-based visual effect evaluation of chevron alignment signs’ colors on drivers through the curves in snow and ice environment”. In: *Journal of Sensors* 2017 (2017).

[16] Tao Wang, Liangjie Xu, Guojun Chen, and Wei Zhao. “A guidence method for lane change detection at signalized intersections in connected vehicle environment”. In: *2019 5th International Conference on Transportation Information and Safety (ICTIS).* IEEE. 2019, pp. 32–38.

[17] Bozhao Qi, Wei Zhao, Xiaohan Wang, Shen Li, and Troy Runge. “A low-cost driver and passenger activity detection system based on deep learning and multiple sensor fusion”. In: *2019 5th International Conference on Transportation Information and Safety (ICTIS).* IEEE. 2019, pp. 170–176.

[18] Wei Zhao, Jiateng Yin, Xiaohan Wang, Jia Hu, Bozhao Qi, and Troy Runge. “Real-time vehicle motion detection and motion altering for connected vehicle: Algorithm design and practical applications”. In: *Sensors* 19.19 (2019), p. 4108.

[19] Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, and Lenka Zdeborová. “Machine learning and the physical sciences”. In: *Reviews of Modern Physics* 91.4 (2019), p. 045002.

[20] Sebastian Raschka. *Python machine learning.* Packt publishing ltd, 2015.

[21] Giuseppe Bonaccorso. *Machine learning algorithms.* Packt Publishing Ltd, 2017.

[22] Wei Zhao, Liangjie Xu, and Bin Sun. “Relationship between vehicle emissions and air quality within a transfer center area of downtown: a case in wuhan, china”. In: *CICTP 2015.* 2015, pp. 3408–3418.

[23] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. “Machine learning basics”. In: *Deep learning* 1 (2016), pp. 98–164.

[24] Batta Mahesh. “Machine Learning Algorithms-A Review”. In: *International Journal of Science and Research (IJSR).[Internet]* 9 (2020), pp. 381–386.

[25] Wei Zhao, Xuan Wang, Bozhao Qi, and Troy Runge. “Ground-level Mapping and Navigating for Agriculture based on IoT and Computer Vision”. In: *IEEE Access* 8 (2020), pp. 221975–221985.

[26] Wenmin Wang, Wei Zhao, Xiaohan Wang, Zhihong Jin, Yuanchen Li, and Troy Runge. “A low-cost simultaneous localization and mapping algorithm for last-mile indoor delivery”. In: *2019 5th International Conference on Transportation Information and Safety (ICTIS).* IEEE. 2019, pp. 329–336.

[27] Taiwo Oladipupo Ayodele. “Types of machine learning algorithms”. In: *New advances in machine learning* 3 (2010), pp. 19–48.

[28] Wei Zhao, William Yamada, Tianxin Li, Matthew Digman, and Troy Runge. “Augmenting Crop Detection for Precision Agriculture with Deep Visual Transfer Learning—A Case Study of Bale Detection”. In: *Remote Sensing* 13.1 (2021), p. 23.

[29] Stephen Marsland. *Machine learning: an algorithmic perspective.* Chapman and Hall/CRC, 2011.

[30] Ethem Alpaydin. *Introduction to machine learning.* MIT press, 2020.

[31] Wei Zhao, Liangjie Xu, Bozhao Qi, Jia Hu, Teng Wang, and Troy Runge. “Vivid: Augmenting vision-based indoor navigation system with edge computing”. In: *IEEE Access* 8 (2020), pp. 42909–42923.

[32] Donald Michie. "“Memo” functions and machine learning”. In: *Nature* 218.5136 (1968), pp. 19–22.

[33] Balas K Natarajan. *Machine learning: A theoretical approach.* Elsevier, 2014.

[34] Hal Daumé. *A course in machine learning.* Hal Daumé III, 2017.

[35] Ka Ho Yuen, Junda Chen, Yue Hu, Ka Wai Ho, A Lazarian, Victor Lazarian, Bo Yang, Blakesley Burkhart, Caio Correia, Jungyeon Cho, et al. “Statistical tracing of magnetic fields: comparing and improving the techniques”. In: *The Astrophysical Journal* 865.1 (2018), p. 54.

[36] A Lazarian, Ka Ho Yuen, Ka Wai Ho, Junda Chen, Victor Lazarian, Zekun Lu, Bo Yang, and Yue Hu. “Distribution of velocity gradient orientations: Mapping magnetization with the velocity gradient technique”. In: *The Astrophysical Journal* 865.1 (2018), p. 46.

[37] Martin Prammer, Suryadev Rajesh, Junda Chen, and Jignesh Patel. “Introducing a Query Acceleration Path for Analytics in SQLite3”. In: *Conference on Innovative Data Systems Research 2022 (CIDR’22)* (2022).