Predicting Garment Worker Productivity

By Ethan Cheung, Moises Jabbour

Table of contents

01

Introduction

What is the garment industry?

The Garment Industry

- Employs over 100 thousand people in the US and 60 million worldwide
- Daily productivity of factories can drastically change from day to day, impacting sales
- Hard to manually predict productivity

Goal: Predict worker productivity using various Machine Learning models

Dataset Used

Number of Attributes	14
Number of Instances	1197
Class Attribute	Day Productivity
Dataset Type	Continuous

Source	UCI Machine Learning Repository - *Productivity Prediction of Garment Employees*.

Attributes

Date	Date in MM-DD-YYYY
Quarter	The corresponding quarter of the month, with each month being split into 4 quarters
Department	Associated department of the instance (sewing or finishing)
Day	Day of the week
Team	Associated team number with the instance (1-12)
Target productivity	Worker productivity assigned by the supervisor (assuming 1 means perfect level production and 0 meaning no production)
SMV	Standard minute value, time allocated to complete a task (basic time + allowance)
WIP	Work in progress, # of unfinished products

Overtime	Amount of overtime by each team in minutes
Incentive	Amount of financial incentive (BDT) to complete a task
Idle time	Amount of time production was interrupted
Idle men	Amount of men idle due to production interruption
No_of_style_change	Number of changes in a particular product's style
No_of_workers	Number of workers in each team

Nominalization of the Class

Advantages

- Simplification of the dataset
- Allows for more accurate prediction of class
- Allows for the use of classifier algorithms

Disadvantages

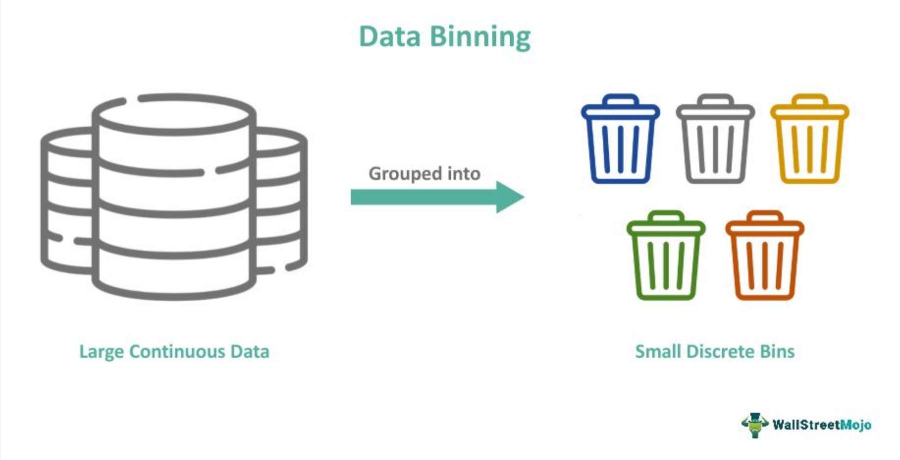
- Reduces the precision of the measurements
- Increases bias, decreases variance

Data Preprocessing

- Handling Missing Values and redundant attributes
- WIP attribute removed (506 missing values, ~42% of the data)
- Removed "date" and "quarter" due to redundancy

Data Preprocessing

- Discretized continuous attributes w/ decimals
- SMV (standard minute value), overtime, bonus incentives, idle men, idle time, and the number of workers
- Binned into equal frequencies



Attribute Selection

Four Algorithms

CorrelationAttributeEval

- Measures the correlation between each attribute and the class
- Pearson's correlation coefficient

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

r = correlation coefficient

x_i = values of the x-variable in a sample

$\bar{x}$ = mean of the values of the x-variable

y_i = values of the y-variable in a sample

$\bar{y}$ = mean of the values of the y-variable

InfoGainAttributeEval

- Evaluates the value of an attribute by measuring the information gained concerning the class
- Information gain is the entropy or the measurement of uncertainty in the dataset

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i)$$

$$IG(Class, Attribute) = H(Class) - H(Class|Attribute)$$

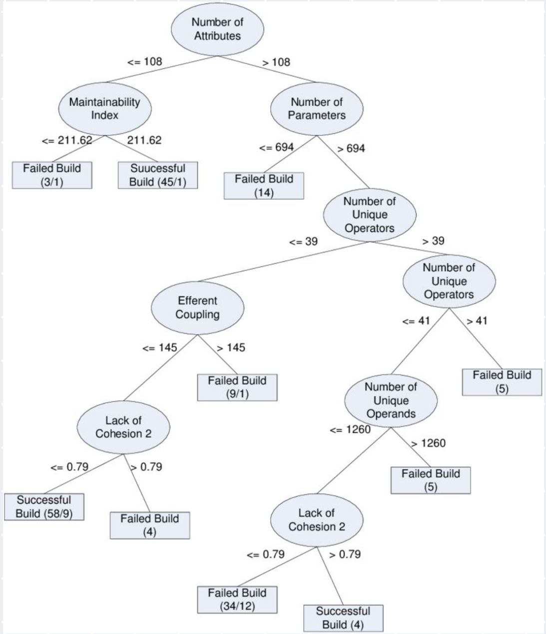
ReliefFAttributeEval

- Evaluates attributes by estimating how well they distinguish between instances that are near each other
- Considers differences between instances of the same label(nearest hit) and those of a different label(nearest miss)

$$W(A_i) = W(A_i) - \frac{1}{m} \sum_{j=1}^{m} (\mathrm{diff}(A_i, I_j, \mathrm{nearestHit}(I_j)) - \mathrm{diff}(A_i, I_j, \mathrm{nearestMiss}(I_j)))$$

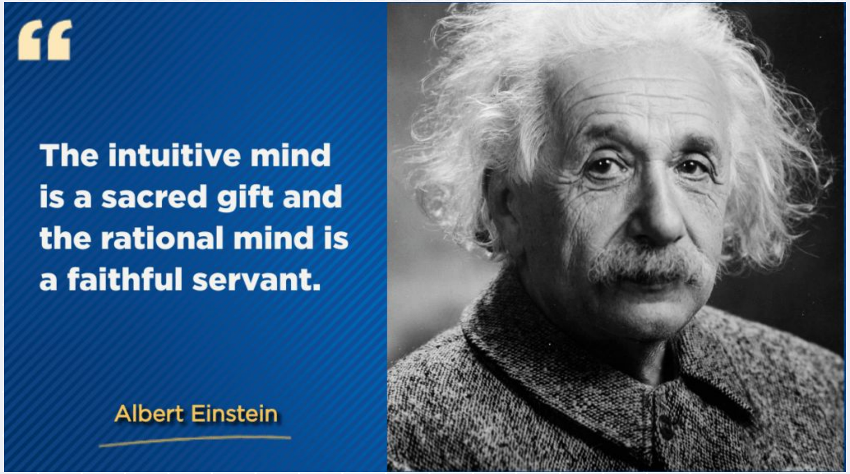
WrapperSubsetEval with J48 Decision Tree

- Evaluates attribute subsets using specific machine learning algorithms, particularly the J48 Decision Tree.
- Uses the performance of the J48 classifier to determine the effectiveness of a subset of attributes in making predictions.
- Constructs a tree using information gain



Intuition

- Removed the quarter and day attribute, lack of implications and effect
- Removed idle time and idle men because they were largely uncontrollable and non-frequent.
- Aimed to have model distinguish more significant attributes if looking at weights



Classifier Algorithms

Four Classifier Models

OneR

- Simple classification algorithm that generates a rule based on a single attribute to classify the data
- Works by checking each attribute for how well it performs in classifying data, and picks the best-performing attribute.

Frequency Tables

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1

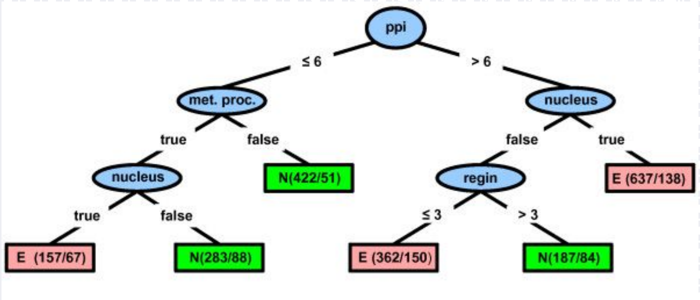
		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3

```
for each predictor P
for each value V of the predictor, generate rule as
find the most frequent class c
create a rule if (P = V) then c
compute the error rate of the rule
select predictor with minimum error rate for its rules
```

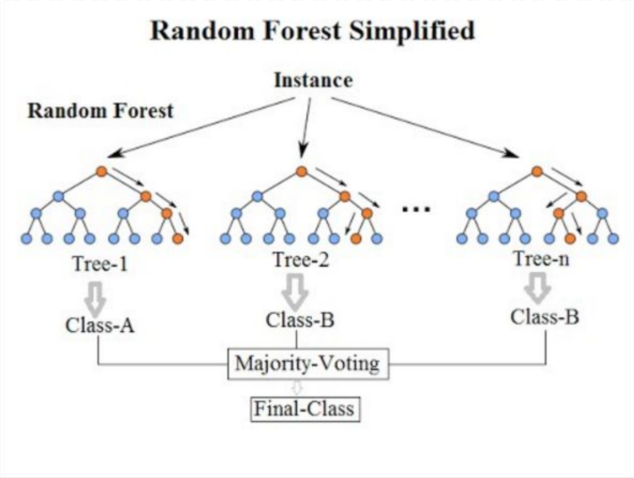
J48 Decision Tree

- Decision tree algorithm that is an extension of 1-R
- Makes a set of rules based on multiple attributes
- Looks at the information gained from each attribute(as seen in the attribute selection section) to divide the dataset into distinct attributes
- Continues until all instances are classified into a single class and no attributes remain.



Random Forest

- Constructs multiple decision trees and outputs the mode of the classes from all the trees
- Builds using a random subset of the training data and a random number of features.



Decision Table

A Sample Decision Table

Condition	Requirement Number				
	1	2	3	4	5
Requester is authorized	F	T	T	T	T
Chemical is available	—	F	T	T	T
Chemical is hazardous	—	—	F	T	T
Requester is trained	—	—	—	F	T
Action					
Accept request			X		X
Reject request	X	X		X	

- Simple classification model that places instances in a table format
- Creates a set of rules based on combinations of attributes for each row
- Performs a lookup on a table and selects the closest matching row.

Results

- K Fold cross validation
- Roughy 80% accuracy
- InfoGainAttributeEval with Random Forest and WrapperSubsetEval with J48
- Four variations of error scores
- Incentive and smv ranked high
- Style changes, targeted_productivity, and day ranked low

	Correlation	Info Gain	Relief	WrapperSubset	Intuition
1-R	76.78%	76.78%	76.78%	76.78%	76.78%
J48	80.28%	80.12%	80.28%	81.62%	80.70%
Random Forest	79.28%	81.62%	80.95%	80.62%	80.79%
Decision Table	79.95%	80.20%	79.95%	79.95%	79.95%

Results Pt 2: Confusion Matrices

	Correlation	Info Gain	Relief	WrapperSubset	Intuition
1-R	830 39 239 89	830 39 239 89	830 39 239 89	830 39 239 89	830 39 239 89
J48	809 60 176 152	809 60 178 150	795 74 162 166	819 50 170 158	794 75 156 172
Random Forest	775 94 154 174	794 75 145 183	774 95 133 195	780 89 143 185	774 95 135 193
Decision Table	816 53 187 141	816 53 184 144	810 59 181 147	810 59 181 147	810 59 181 147

Results Pt 3: Attribute Selection

Correlation	Info. Gain	Relief	Wrapper
0.27416 8 incentive 0.24777 2 department 0.13956 9 idle_time 0.13956 10 idle_men 0.1369 12 no_of_workers 0.09686 6 smv 0.08654 4 team 0.06862 7 over_time 0.03666 3 day 0.03012 1 quarter 0.00633 5 targeted_productivity 0.00142 11 no_of_style_change	0.12903 8 incentive 0.10893 6 smv 0.07992 12 no_of_workers 0.06531 7 over_time 0.04406 2 department 0.02636 4 team 0.01211 9 idle_time 0.01211 10 idle_men 0.00553 1 quarter 0.00543 3 day 0 11 no_of_style_change 0 5 targeted_productivity	0.21746 6 smv 0.1929 8 incentive 0.13033 7 over_time 0.11228 12 no_of_workers 0.06867 1 quarter 0.0533 4 team 0.02804 5 targeted_productivity 0.02135 11 no_of_style_change 0.00819 2 department 0.00301 9 idle_time 0.00301 10 idle_men -0.02005 3 day	Selected Attributes: quarter department Day targeted_productivity smv over_time incentive no_of_workers productivity_met

Conclusion

- InfoGainAttributeEval with Random Forest
- Future projects to examine specific attribute weights
- Potentially use a deep neural network

```
=== Summary ===

Correctly Classified Instances         977               81.6207 %
Incorrectly Classified Instances       220               18.3793 %
Kappa statistic                          0.5052
Mean absolute error                      0.2536
Root mean squared error                  0.3563
Relative absolute error                 63.7075 %
Root relative squared error             79.8933 %
Total Number of Instances             1197

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.914    0.442    0.846      0.914    0.878      0.512    0.862     0.938     yes
                 0.558    0.086    0.709      0.558    0.625      0.512    0.862     0.724     no
Weighted Avg.    0.816    0.345    0.808      0.816    0.809      0.512    0.862     0.879

=== Confusion Matrix ===

   a   b   <-- classified as
 794  75 |   a = yes
 145 183 |   b = no
```

References

Al Imran, Abdullah, Md Shamsur Rahim, and Tanvir Ahmed. "Mining the Productivity Data of the Garment Industry." International Journal of Business Intelligence and Data Mining, vol. 19, no. 3, 2021, pp. 319–342. Inderscience Enterprises Ltd.

Hasan, Mohammad Shahadat, et al. "An Approach to Predict Employee Productivity Using a Hybrid CNN-LSTM Model." 2021 International Conference on Artificial Intelligence (ICAI), IEEE, 2021, pp. 1-7, https://ieeexplore.ieee.org/document/9689701. Accessed 20 Oct. 2024.

Schlattmann, Maximilian. "OneR: Machine Learning in Under One Minute." Ephorie Blog, 13 Mar. 2020, https://blog.ephorie.de/oner-machine-learning-in-under-one-minute. Accessed 20 Oct. 2024.

Frank, Eibe, et al. The WEKA Workbench. Version 3.8, University of Waikato, 2023, https://www.weka.io/?utm_medium=cpc&utm_ Accessed 20 Oct. 2024.

Thanks!

Do you have any questions?